United States Court of Appeals for the Federal Circuit

04-1009

BUSINESS OBJECTS, S.A.,

Plaintiff-Appellant,

v.

MICROSTRATEGY, INC.,

Defendant-Appellee.

Daniel J. Furniss, Townsend and Townsend and Crew, LLP, of Palo Alto, California, argued for plaintiff-appellant. Of counsel on the brief were Joseph A. Greco and Gregory S. Bishop.

Peter E. Moll, Howrey Simon Arnold & White, LLP, of Washington, DC, argued for defendant-appellee. With him on the brief were James F. Valentine and Matthew E. Hocker of Menlo Park, California; and Richard L. Stanley of Houston, Texas.

Appealed from: United States District Court for the Northern District of California

Judge Charles R. Breyer

United States Court of Appeals for the Federal Circuit

04-1009

BUSINESS OBJECTS, S.A.,

Plaintiff-Appellant,

v.

MICROSTRATEGY, INC.,

Defendant-Appellee.

DECIDED: January 6, 2005

Before RADER, SCHALL, and PROST, Circuit Judges.

RADER, Circuit Judge.

The United States District Court for the Northern District of California entered summary judgment that MicroStrategy, Inc.'s products do not infringe claims 1, 2 and 4 of Business Objects' U.S. Patent No. 5,555,403 (issued Sept. 10, 1996) ('403 patent) either literally or under the doctrine of equivalents. Business Objects, S.A. v. MicroStrategy, Inc., 280 F. Supp. 2d 1000 (N.D. Cal. 2003) (Summary Judgment Order). Because the district court did not err in construing the asserted claims, this court affirms the district court's ruling of no literal infringement of claims 1, 2 and 4. Because the district court also did not err in holding that prosecution history bars Business Objects from asserting equivalents for claims 1 and 2, this court affirms that portion of the order as well. Because the district court erred in holding that an amendment to claim 4 during prosecution narrowed the scope of the claim, this court reverses the district court’s ruling that Business Objects is precluded from claiming equivalents of claim 4. Finally, this court vacates the district court’s dismissal of MicroStrategy’s counterclaims as moot, and remands the case for a determination of infringement of claim 4 under the doctrine of equivalents and of MicroStrategy’s counterclaims consistent with this opinion.

I.

Business Objects owns he ’403 patent, which claims an improvement for searching relational databases. A relational database is a computerized compilation of data organized into tables, each table having columns (attributes), with column headings, and rows of information. Tables that share at least one attribute in common are “related.” Tables without a common attribute may still be related via other tables with which they do share a common attribute. The pathways relating those separate tables to each other are called “joins.” Once tables have been related by a join, a user may combine or correlate the information in the joined tables to derive new useful information.

Users access and correlate information in relational databases only by use of a relational database management system (RDBMS), which consists of hardware and software. To access such information, a user sends queries to the RDBMS, which executes the queries and retrieves the requested information from the tables in the relational database. A RDBMS, however, only recognizes queries written in complex “query languages.” The most common query language is Structured Query Language (SQL).

A proper query in these languages consists of one or more “clauses.” Common types of clauses are SELECT, WHERE, FROM, HAVING, ORDER BY, and GROUP BY clauses**.** Thus, to compose a proper inquiry, a user must understand the structure and content of the relational database as well as the complex syntax of the specific query language. These complexities generally prevent laypersons from drafting queries in query languages.

The ’403 patent claims a method that allows end users to query a relational database without knowing a query language or understanding the structure of the relational database. The method employs a “manager” (a skilled human operator who defines the business objects with knowledge of the database) that sets the parameters to allow lay users to retrieve information from a relational database. The manager achieves this objective by creating a new “universe.” This universe is a user-friendly representation of the contents of a relational database that are relevant to the lay users. A universe consists of “business objects,” “classes,” “joins” and “contexts.”

A business object consists of a familiar name, elements of a SELECT clause, and elements of a WHERE clause. The familiar name is a common word that the lay user recognizes as designating the information for retrieval, such as “Sales.” The elements of the SELECT and WHERE clauses associated with the familiar name are not visible to the user, but are used by a “query engine” to generate the appropriate query language for execution by the RDBMS.

A class is merely a set of logically-related business objects with certain predetermined attributes in common. A class of business objects may be presented to a lay user, such as in a list or drop-down menu.

As explained above, joins specify the path relating two or more tables to each other. When more than one join exists between two tables, the context specifies which join the query engine will incorporate into the query to retrieve the desired information. The manager may set a default context when creating the universe, or the program may ask the end user to specify the context.

After the manager has created a universe, lay users may work with common language to query the relational database. To make a query, a lay user needs to merely: (1) select the familiar names of business objects that the user desires to correlate; (2) specify any desired conditions to limit the results; and (3) state the retrieval order for the data. The query engine then generates a complete query in the appropriate query language, using the proper clauses, joins, and syntax, for execution by the RDBMS. This opinion differentiates between SELECT and WHERE clauses associated with business objects and the SELECT and WHERE clauses generated by the query engine by designating the clauses generated by the query engine as "SELECT and WHERE statements." The final query generated by the query engine includes SELECT and WHERE statements that consist of the SELECT and WHERE clauses associated with each business object.

The invention allows the lay user to bring different business objects together. Thus, a user may use the same business object to obtain different information depending upon which other business objects are included in the query. For example, the business object "Sales" connected to the business object "Customers" returns the dollar amount sold to each customer. The same business object "Sales" connected to the business object "Product" returns the total revenue generated by each product.

Thus, the meaning of the information returned by the "Sales" business object is dynamic, changing according to its association with other business objects in the query. The concept that the same business object may return different information is called "dynamic semantics."

Business Objects asserts claims 1, 2 and 4 of the '403 patent. Claim 1, from which claim 2 depends, states:

> A method for accessing values in a relational database, wherein the relational database operates in a computer system and provides returned values responsive to queries specified in a predefined query language, wherein the relational database supports the use of functions and operators to perform operations on values within the database, wherein the relational database includes a plurality of tables, wherein each table is associated with one or more attributes, wherein each attribute has a set of values, wherein the method includes a user interface executing on a computer system operated by a human user, wherein the computer system executing the user interface includes a processor coupled to a memory, wherein the processor is further coupled to the user interface and the relational database, the method comprising the following 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.

'403 patent, col. 16, l. 39 – col. 17, l. 2 (emphasis added).

Claim 4 states:

> A relational database access system comprising:
>
> a computer system including a processor coupled to a memory;

> a relational database including attributes, wherein the relational database is responsive to a query in a predefined query language for accessing the attributes in the relational database;
>
> a user interface for allowing a human user to specify one or more familiar names in the form of a user query;
>
> 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; and
>
> query engine means for <u>generating queries in the predetermined query language</u> based on a given combination of two or more selected and associated familiar names, wherein the query engine means generates 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.

’403 patent, col. 17, l. 65 – col. 18, l. 23 (emphasis added).

MicroStrategy’s accused products also allow lay users to use familiar names to query a relational database. The accused products, however, use a much more sophisticated approach to generating queries than the invention of the ’403 patent. The accused products have three separate “hierarchies” of information and “entities” that are derived from the physical tables in the relational database. The first hierarchy is the “abstraction” layer, which creates new tables (abstraction tables) that correspond to the tables in the physical database. These abstraction tables serve to make all higher-level MicroStrategy metadata independent of the stringent form and syntax requirements of the tables in the physical database. A person, like the manager in the ’403 patent, designates these abstraction tables. The abstraction tables may contain the same information as the tables of the physical database, or may contain information from the physical database.

The next hierarchy is the “Schema” level. In this level, new tables are created (schema tables) using “FactInfo” and “AttributeInfo.” The schema tables do not need to correlate on a one-to-one basis with the abstraction tables, and may combine information from different abstraction tables. When creating the schema tables, the manager also creates a list of the facts and a list of the attributes in each schema table. Any particular fact or attribute may appear in multiple schema tables.

The third hierarchy is the “Application” layer. Lay users utilize this layer to assemble queries. The lay user makes “reports” using “templates,” “filters” and “metrics.” A report is the specification of a lay user and is comprised of a template and a filter. The template specifies the metrics and attributes to be included in the report, and the filter describes conditions on data to be included in the report. A metric is a mathematical compilation of lower level entities. Importantly, none of the reports, filters, templates, metrics, attributes or facts is tied to any specific table of the physical database but instead each is associated with multiple schema tables.

Once the end user specifies a report, the query engine scans the lists of facts and attributes called for by the report to determine which group of tables will make the “Best Base Table Set,” i.e., the most efficient group of schema tables from which to generate the report. The query engine identifies each schema table containing a required fact or attribute. The query engine next selects the best set of tables from the identified tables, using various criteria, and eliminates the unnecessary schema tables.

After selecting the schema tables for the query, the query engine constructs the most efficient joins between the pertinent tables, and decides whether multiple queries must be run to obtain the results for the report. The query engine next generates a query in “canonical” format for each pathway or join tree. The generation of canonical format queries entails reducing the names of the schema tables and the names of the columns therein to the names of the physical tables and columns in the relational database. Finally, the query engine translates the canonical query into the specific query language the RDBMS requires, incorporating the specific form and syntax of that query language.

II.

This court reviews a grant of summary judgment without deference to the decision of the district court. Anderson v. Liberty Lobby, Inc., 477 U.S. 242, 255 (1986); Johns Hopkins Univ. v. CellPro, Inc., 152 F.3d 1342, 1353 (Fed. Cir. 1998). Summary judgment is appropriate only where “there is no genuine issue as to any material fact and . . . the moving party is entitled to judgment as a matter of law.” Fed. R. Civ. P. 56(c); see also Celotex Corp. v. Catrett, 477 U.S. 317, 322 (1986). “On summary judgment, the evidence must be viewed in the light most favorable to the party opposing the motion, with doubts resolved in favor of the nonmovant.” Crown Operations Int’l, Ltd. v. Solutia Inc., 289 F.3d 1367, 1375 (Fed. Cir. 2002) (citations omitted).

A court determines patent infringement by construing the claims and then applying that construction to the accused process or product. Markman v. Westview Instruments, Inc., 52 F.3d 967, 976 (Fed. Cir. 1995) (en banc), aff’d, 517 U.S. 370 (1996). This court reviews claim construction without deference. Cybor Corp. v. FAS Techs., Inc., 138 F.3d 1448, 1454 (Fed. Cir. 1998) (en banc). “In the context of summary judgment, this court reviews the second determination for genuine disputes of material facts that would preclude a grant of summary judgment.” AFG Indus. v.

Cardinal IG Co., 375 F.3d 1367, 1371 (Fed. Cir. 2004) (citing Hilgraeve Corp. v. McAfee Assocs., Inc., 224 F.3d 1349, 1352-54 (Fed. Cir. 2000)). Thus, MicroStrategy is entitled to summary judgment of no infringement only if the facts and inferences, when viewed in the light most favorable to Business Objects, would not persuade a reasonable jury to return a verdict in favor of Business Objects, the non-moving party. Anderson, 477 U.S. at 255. Finally, the prosecution history estoppel limit on the doctrine of equivalents presents a question of law, which this court reviews without deference. See Glaxo Wellcome, Inc. v. Impax Labs., Inc., 356 F.3d 1348, 1351 (Fed. Cir. 2004) (citing Wang Labs. v. Mitsubishi Elecs. Am., 103 F.3d 1571, 1578 (Fed. Cir. 1997)).

A. Claim Construction

The district court construed the "associating step" limitation of claim 1 as associating a familiar name with "elements of SELECT and WHERE clauses" before the "generating a query" step. Business Objects, S.A. v. MicroStrategy, Inc., No. C 01-3908 CRB, slip op. at 9 (N.D. Cal. May 1, 2003) (Construction Order). Business Objects does not challenge this claim construction, but argues that the district court improperly narrowed the meaning of "elements" in the Summary Judgment Order. See 280 F. Supp. 2d at 1003-04 (comparing the accused products with the associating step limitation). The Summary Judgment Order states that "'an element' is not so broad as to encompass information that will be, but is not yet, associated with a SELECT/WHERE clause." Id.

Although affirming the district court's construction of the associating step limitation, this court restates the claim construction in the terminology adopted for this opinion to clarify that claim construction. The associating step of claim 1 requires the association of a familiar name with SELECT/WHERE clauses used by the query engine to generate the SELECT/WHERE statements. Thus, the phrase "element of a SELECT/WHERE clause" in the district court's Construction Order refers to an element of the final SELECT/WHERE statement, as labeled in this opinion, generated by the query engine. The "element" is the SELECT/WHERE clause associated with a familiar name that the query engine concatenates with the SELECT/WHERE clauses of other familiar names to generate SELECT/WHERE statements for execution by the RDBMS. The district court's construction of the associating step limitation, and subsequent clarification in the Summary Judgment Order that "an element" excludes "information that will be, but is not yet, associated with a SELECT or WHERE clause[,]" therefore, are correct.

This court rejects Business Objects' argument that this construction of the associating step precludes dynamic semantics. The '403 patent clearly states 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 a column header. The association of two Business Objects will produce an answer made of two columns which contents depend on the association." '403 patent col. 8, ll. 43-49. Thus, the '403 patent clearly teaches that the inclusion of a particular familiar name in a query will always return information from the same table and column, but the information returned by using the familiar name will change depending on the other familiar names included in the query. For example, assume that the clause "SELECT: customers.cust_name" is associated with the familiar name Customers. See '403 patent col. 6, ll. 21-44. Running a query with the familiar name Customers by itself returns the names of all customers in the customers.cust_name table and column. The same familiar name Customers combined with a familiar name "Loans" returns the names of all customers in the customers.cust_name table and column that have outstanding loans. The same familiar name Customers combined with a familiar name "Trucks" returns the names of all customers in the customers.cust_name table and columns who have purchased trucks. The names of customers returned by each of the foregoing queries will be different, but in each case the names are retrieved from the "customers.cust_name" table and column. Accordingly, associating the SELECT clause with the familiar name Customers does not preclude the list of names returned from changing dynamically according to the combination of "Customers" with other familiar names in a query. Consequently, the district court's claim construction requiring the association of SELECT/WHERE clauses to familiar names before the "generating a query" step does not preclude dynamic semantics.

Business Objects also challenges the district court's construction of the terms "predefined query language" and "predetermined query language." Absent any party dispute, this court construes these two terms as having the same meaning. The district court initially construed these terms to mean that a "query language that must be determined prior to the 'generating a query' step, but not necessarily prior to the associating step. Furthermore, the 'predefined [or predetermined] query language' must support the functions and operators contained in the associating step's SELECT clause." Construction Order at 9-10. Business Objects disputes the portion of this claim construction requiring the predetermined language to support the functions and operators contained in the associating step's SELECT clause.

Business Objects argues that the district court's construction improperly imports the "associating step" from claim 1 into claim 4. To the contrary, this court agrees with both parties that the "query engine means" limitation of claim 4 is a means-plus-function limitation governed by 35 U.S.C. § 112, ¶ 6. This court also agrees with the parties that the corresponding structure consists of the 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. One step of this algorithm is the generation of the SELECT statements by concatenating SELECT clauses associated with familiar names. '403 patent, col. 10, ll. 6-16. Because the query engine means of claim 4 must concatenate SELECT clauses associated with familiar names to generate SELECT statements in the predetermined query language, the predetermined query language must support the SELECT clauses associated with the familiar names. The district court's claim construction, therefore, is consistent with the structural limitations of the query engine means and does not improperly import the associating step limitation of claim 1. Consequently, this court affirms the district court's construction of the terms "predetermined query language" and "predefined query language."

B. Literal Infringement

Business Objects' challenge of the district court's grant of summary judgment of no literal infringement of claims 1 and 2 rests on its argument that the district court's construction of the associating step is erroneous. Having affirmed the district court's construction of the associating step of claim 1, this court also affirms the district court's grant of summary judgment of no literal infringement of claims 1 and 2 of the '403 patent because "the accused products do not associate a familiar name with a SELECT and

WHERE clause prior to the generation of a query" as required in the associating step of claim 1. Summary Judgment Order, 280 F. Supp. 2d at 1003-04.

Business Objects challenges the district court's grant of summary judgment of no literal infringement of claim 4 by arguing that the district court erroneously construed claim 4 to associate a SELECT clause with a familiar name. This court affirmed the district court's holding that claim 4 requires that association. Because the parties do not dispute that the accused devices do not associate a SELECT clause with a familiar name, this court affirms the district court's finding of no literal infringement of claim 4.

B. Infringement under the Doctrine of Equivalents

Infringement under the doctrine of equivalents occurs when a claimed limitation and the accused product perform substantially the same function in substantially the same way to obtain substantially the same result. See Warner-Jenkinson Co. v. Hilton-Davis Chem. Co., 520 U.S. 17, 40 (1997). However, the Supreme Court in Festo Corp. v. Shoketsu Kinzoku Kogyo Kabushiki Co., 535 U.S. 722, 736 (2002) (Festo VIII), ruled that "a narrowing amendment made to satisfy any requirement of the Patent Act may give rise to an estoppel" on infringement by equivalents. A narrowing amendment gives rise to a presumption that the patentee surrendered any subject matter between the amended claim as originally filed and claims allowed by the Patent and Trademark Office (PTO). Id. at 741. To overcome this presumption, a patentee may show "that the alleged equivalent could not reasonably have been described at the time the amendment was made, or that the alleged equivalent was tangential to the purpose of the amendment, or that the equivalent was not foreseeable (and thus not claimable) at the time of the amendment." Glaxo Wellcome, 356 F.3d at 1351-52 (citing Festo VIII, 535 U.S. at 740-41).

The district court applied prosecution history estoppel to prevent Business Objects from asserting equivalents of the associating step limitation of claims 1 and 2. Summary Judgment Order, 280 F. Supp. 2d at 1006. Business Objects concedes that the inclusion of the associating step limitation to claim 1 was the result of a narrowing amendment to advance patentability during prosecution of the ’403 patent. Thus, this amendment erects a Festo presumption of estoppel. Business Objects attempts to overcome that presumption by arguing that the amendment is only tangentially related to MicroStrategy’s accused equivalent. Business Objects, however, argued to the examiner at the PTO that the addition of the associating step to claim 1 rendered claim 1 allowable over two prior art references because neither of those references associated WHERE clauses with a familiar name. Business Objects now alleges that the accused products infringe claims 1 and 2 under the doctrine of equivalents because they associate something equivalent to SELECT and WHERE clauses to the familiar name. Thus, the district court correctly discerned that Business Objects’ amendments "are directly related to the claimed equivalents." Id. This court affirms the district court’s holding that Business Objects may not assert that the accused products contain equivalents of the associating step in claims 1 and 2.

Business Objects also challenges the district court’s holding that it is estopped "from contending that the accused devices practice a function equivalent to the ’403 query engine means." Id. at 1007. Claim 4 was added during the prosecution of the ’403 patent. Claim 4 was similar to the original independent system claims filed, but changed, inter alia, the language "translating said user query into a structured query language (SQL) equivalent statement," to "generating queries in the predetermined query language". The district court held that this amendment narrowed the scope of the function of the query engine means for purposes of patentability and raised a Festo presumption. Id. at 1006-07.

To the contrary, the amended term "generating queries in a predetermined query language" is broader than the original term "translating said user query into a structured query language (SQL) equivalent statement". The original term "translating said user query into a structured query language (SQL) equivalent statement" means to translate the query specified by the lay user in terms of selected business objects into its SQL equivalent. The specification at column 10, lines 10-13, conveys the same meaning of the language: "The QE (Query Engine) replaces the Business Objects select [sic] in the Result Objects window by their SQL equivalent. Each Business Object has an SQL equivalence, which can be any SQL Select clause compatible string." This court rejects MicroStrategy's argument that "SQL equivalent" refers to languages that are equivalent to SQL. The specification does not discuss or refer to languages that are "equivalent" to SQL. Moreover, MicroStrategy cannot point to any other indicia in the record that would support its proposed interpretation of the term.

The parties do not dispute that the ordinary meaning of "query language" includes both SQL and query languages other than SQL. MicroStrategy contends that regardless of whether "query language" is broader than "SQL," the qualifier "predetermined" narrows the function of the query engine means as originally filed. This argument is not persuasive. By designating SQL as the language into which the business objects are to be translated, the original term implicitly "predetermined" that the query language would be SQL. Because the "predetermined" limitation was implicitly contained in the original term, the amendment did not narrow the scope of the query engine means by expressly stating that the query language must be "predetermined" in the amended term. Business Objects, therefore, did not narrow this limitation for purposes of patentability and is not precluded from claiming equivalents of the query engine means in the accused products. See Festo Corp. v. Shoketsu Kinzoku Kogyo Kabushiki Co., 344 F.3d 1359, 1366 (Fed. Cir. 2003) (en banc) ("If the amendment was not narrowing, then prosecution history estoppel does not apply."), reh'g en banc denied, No. 95-1066, 2003 U.S. App. LEXIS 24135 (Fed. Cir. Nov. 5, 2003), cert. denied, 124 S. Ct. 2018-19 (2004). This court does not address whether Business Objects may be estopped from claiming equivalents to limitations of claim 4 other than the "predefined query language" limitation addressed in this opinion because those issues were not addressed below nor raised by the parties on appeal.

III.

This court affirms the district court's construction of the "associating step" and "predefined query language" limitations. This court affirms the district court's finding of no literal infringement of claims 1, 2 and 4 and that Business Objects is estopped from claiming equivalents of the "associating step" in claims 1 and 2. This court vacates the district court's finding that Business Objects' amendment of the "predefined query language" estops it from claiming equivalents of the function of the query engine means in claim 4. This court remands this case for a determination of infringement of claim 4 under the doctrine of equivalents consistent with this opinion and reinstates MicroStrategy's counterclaims because they are no longer moot.

Having addressed the merits of the district court's judgment, we need not address the district court's denial of Business Objects' motion to vacate the judgment.

COSTS

Each party shall bear its own costs.

AFFIRMED IN PART, VACATED IN PART, and REMANDED