

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SERVICENOW, INCORPORATED,
Petitioner,

v.

INQUISIENT INCORPORATED,
Patent Owner.

IPR2023-01073
Patent 8,224,855 B2

Before BRIAN J. McNAMARA, GREGG I. ANDERSON, and
CHARLES J. BOUDREAU, *Administrative Patent Judges*.

ANDERSON, *Administrative Patent Judge*.

DECISION
Denying Institution of *Inter Partes* Review
35 U.S.C. § 314

I. INTRODUCTION

ServiceNow, Inc. (“Petitioner”) filed a petition,¹ Paper 6 (“Petition” or “Pet.”), to institute an *inter partes* review (“IPR”) of claims 1–33 (the “challenged claims”) of U.S. Patent No. 8,224,855 B2 (“the ’855 patent”) under 35 U.S.C. § 311. InQuisient, Inc. (“Patent Owner”) filed a Preliminary Response, Paper 7 (“Prelim. Resp.”), contending that the Petition should be denied as to all challenged claims. We authorized Petitioner’s filing of a Reply to Patent Owner’s Preliminary Response (Paper 8, “Pet. Prelim. Reply”) and Patent Owner’s filing of a Preliminary Sur-reply (Paper 9, “PO Prelim. Sur-reply”).

We have jurisdiction under 35 U.S.C. § 6 (2018). This Decision on Institution is issued pursuant to 35 U.S.C. § 314, which provides that an *inter partes* review may not be instituted unless the information presented in the Petition “shows that there is a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the claims challenged in the petition.”

Having considered the arguments and the associated evidence presented in the Petition, the Preliminary Response, Petitioner’s Reply to Patent Owner’s Preliminary Response, and Patent Owner’s Preliminary Sur-reply, for the reasons described below, we decline to institute *inter partes* review.

II. REAL PARTIES IN INTEREST

Petitioner identifies itself (ServiceNow, Inc.) as its real party-in-interest. Pet. 69.

¹ The “Petition” corrects a first Petition filed as Paper 2.

Patent Owner identifies itself (InQuisient, Inc.) as its real party-in-interest. Paper 5, 1.

III. RELATED MATTERS

The parties state that the '855 patent is asserted in the following litigation: *InQuisient Inc. v. ServiceNow, Inc.*, 1-22-cv-00900 (D. Del.) (“Delaware Lawsuit”). Pet. 69; Paper 5, 1.

Petitioner also identifies U.S. Patent Nos. 7,979,468 B2 and 8,219,585 that Petitioner challenges in *ServiceNow v. InQuisient, Inc.*, IPR2023-01071 and IPR2023-011072, respectively, as patents related to the '855 patent.

IV. THE '855 PATENT

The '855 patent relates to systems and methods for storing, managing, indexing, interrelating and/or retrieving data sets. Ex. 1001, 1:15–18. The '855 patent states that a conventional system implements its own data model and rules according to a user interface, requiring developers to write dedicated code and user interface code that displays or manages data. *Id.* at 1:22–28. The '855 patent seeks to provide a system and method to store, manage, index, interrelate, and/or retrieve data sets in a manner that is independent of the data model. *Id.* at 1:52–54. The system includes a processor and data repository. *Id.* at 1:61–62. An exemplary entity-relationship diagram of a elements metadata repository (EMR) shown in Figure 1 includes eight modules recited in claim 1. *Id.* at 3:3–4. A database creator enters database information in the modules rather than writing dedicated code and user interface code. *Id.* at 8:55–65. In this elements-based approach, each of the elements conforms to rules of generalized things in the universe, e.g., through entries in an element relation table. *Id.* at 9:24–29.

The challenged claims recite a data repository that includes an element module configured to store and uniquely identify a plurality of elements each identified by a unique element identification. Ex. 1001, 2:1–3, 6:13–35 (describing elements as uniquely identified by an ID, name, and class identifier that points to an indicator of the type or classification of the element). The EMR places “things” into an element table—an element stored in an element table implies that the user can find similarity between any “thing” the user would like to discuss, report, store, or manage, without regard to their exact nature. *Id.* at 6:2–10.

The data repository also includes an element relation module configured to store one or more relationships between the elements in the element module. Ex. 1001, 2:4–6, 6:35–45 (describing an element relation table that stores relationships between elements, e.g., each entry in the element relation table contains a pointer to at least two elements in the elements table and a type definition describing the type of relationship).

A class module is configured to define at least one class of the elements and store the class. Ex. 1001, 1:66–2:7. The EMR also includes an attribute module configured to define and store the attributes and a class attribute module configured to define and store one or more class-attribute associations between at least one of the attributes and the class, and a type definition module configured to define and store one or more types of the class, the attributes related to the class, and the relationships between the elements. *Id.* at 2:6–12. An attribute table 110 holds the definition of each attribute, e.g., the physical name and type of data that corresponds to the database column name and column type where the actual vales are stored. *Id.* at 5:4–8. A class attribute table may associate a given attribute with one or more classes in a class table. *Id.* at 5:9–10.

A type definition module is configured to define and store one or more types of the class, the attributes related to the class and the relationship between the elements. Ex. 1001, 2:10–14.

The EMR can utilize entries in a named text table to name structured query language (SQL) queries and filter query entries into one or more dynamic folders, such that the result of a query may vary over time. Ex. 1001, 10:64–11:39.

V. ILLUSTRATIVE CLAIMS

Claim 1, reproduced below with paragraph designations used in the Petition. Pet. v. Claim 1 is illustrative of the subject matter of the '855 patent and is reproduced below.

1[pre] A computerized system for manipulating data sets, the computerized system comprising:

[a] a hardware-based processor; and

[b] a data repository that contains the data sets and one or more layers of metadata of the data in the data sets, comprising:

[c] a class module configured to define at least one class of elements;

[d] an attribute module configured to define one or more attributes:

[e] a class attribute module configured to define and store one or more class-attribute associations between at least one of the attributes and the class;

[f] an element attribute module for the at least one class configured to: define and store the attributes of the elements in the class; and transmit the attributes and the elements;

- [g] an element module configured to receive the transmitted attributes and elements from the element attribute module, store the attributes and elements, and uniquely identify the elements, wherein the elements are identified by unique element identifications;
- [h] an element relation module configured to store one or more relationships between the elements in the element module;
- [i] an element history module configured to store one or more indicators of a modification to the elements; and
- [j] an element document module configured to store formatting data for grouping a subset of the elements.

Ex. 1001, 16:22–53; Pet. v. Independent claim 12 recites a method with limitations generally corresponding to those of claim 1. Ex. 1001, 17:30–53. Independent claim 23 recites a non-transitory computer readable medium with limitations generally corresponding to those of claim 1. *Id.* at 18:32–60.

VI. ASSERTED GROUNDS

Petitioner asserts that claims 1–33 would have been unpatentable on the following grounds:

Claim(s) Challenged	35 U.S.C. § ²	Reference(s)
1, 3–12, 14–23, 25–33	103(a)	Althoff, ³ Surjanto ⁴
2, 13, 24	103(a)	Althoff, Surjanto, Nye ⁵

² The Leahy-Smith America Invents Act (“AIA”) included revisions to 35 U.S.C. § 103 that became effective on March 16, 2013. Because the ’855 patent issued from an application filed before March 16, 2013, we apply the pre-AIA version of § 103.

³ U.S. Patent No. 6,374,252 B1; Apr. 16, 2002 (Ex. 1006).

⁴ XLM Content Management Based on Object-Relation Database Technology, June 2000 (Ex. 1008).

⁵ U.S. Patent No. 6,341,279 B1; Jan. 22, 2002 (Ex. 1005).

Claim(s) Challenged	35 U.S.C. § ²	Reference(s)
4, 6–7, 15, 17–18, 26, 28–29	103(a)	Althoff, Surjanto, Moore ⁶

VII. LEVEL OF ORDINARY SKILL IN THE ART

Petitioner describes a person of ordinary skill (“POSITA” or “ordinarily skilled artisan”) as one having “a Bachelor’s degree in Computer Science and three to five years of industry experience in application development, database development, enterprise application management; or a Ph.D. specializing in the field.” Pet. 2–3 (citing Ex. 1003, Declaration of Dr. Jules White ¶ 23). Petitioner further cites Dr. White as stating that more education could compensate for less experience and vice versa. *Id.* The Patent Owner Preliminary Response does not directly address the level of ordinary skill.

The level of ordinary skill in the art may be evidenced by the references themselves. *See Okajima v. Bourdeau*, 261 F.3d 1350, 1355 (Fed. Cir. 2001); *In re GPAC Inc.*, 57 F.3d 1573, 1579 (Fed. Cir. 1995); *In re Oelrich*, 579 F.2d 86, 91 (CCPA 1978). As Petitioner’s unchallenged description of the level of ordinary skill appears appropriate to the subject matter of the ’855 patent and the references, we apply it for purposes of this Decision.

VIII. CLAIM CONSTRUCTION

We interpret claim terms using “the same claim construction standard that would be used to construe the claim in a civil action under 35 U.S.C. 282(b).” 37 C.F.R. § 42.100(b). In this context, claim terms “are generally given their ordinary and customary meaning” as understood by a person of

⁶ U.S. Patent No. 7,627,552 B2; Dec. 1, 2009 (Ex. 1007).

ordinary skill in the art in question at the time of the invention. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312–13 (Fed. Cir. 2005) (citations omitted) (en banc). “In determining the meaning of the disputed claim limitation, we look principally to the intrinsic evidence of record, examining the claim language itself, the written description, and the prosecution history, if in evidence.” *DePuy Spine, Inc. v. Medtronic Sofamor Danek, Inc.*, 469 F.3d 1005, 1014 (Fed. Cir. 2006) (citing *Phillips*, 415 F.3d at 1312–17). Extrinsic evidence is “less significant than the intrinsic record in determining ‘the legally operative meaning of claim language.’” *Phillips*, 415 F.3d at 1317 (citations omitted).

Any special definition for a claim term must be set forth in the specification with reasonable clarity, deliberateness, and precision. *In re Paulsen*, 30 F.3d 1475, 1480 (Fed. Cir. 1994).

We construe only those claim terms that require analysis to determine whether to institute *inter partes* review. *See Realtime Data, LLC v. Iancu*, 912 F.3d 1368, 1375 (Fed. Cir. 2019) (“The Board is required to construe ‘only those terms . . . that are in controversy, and only to the extent necessary to resolve the controversy’” (quoting *Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999))).

A. Module

Noting that the ’855 patent Specification describes a data repository as including modules, Petitioner contends that a person of ordinary skill would have understood a “‘module’ to be a portion of a data repository.” Pet. 3. Petitioner notes that in the Delaware Lawsuit, Patent Owner conceded “an example of a ‘module’ is ‘a table or a portion of a table’ (i.e., a data

repository).” *Id.* (citing Ex. 1013,⁷ 78). Petitioner then states that it “applies Patent Owner’s own definition of a ‘module’ throughout this Petition.” *Id.*

Patent Owner argues that, in the context of the ’855 patent, “the term ‘module’ would be understood to be a table-based structure.” Prelim. Resp. 17. Further, Patent Owner contends “a module comprises one or more columns within one or more tables.” *Id.* According to Patent Owner, to the extent the panel issues a formal construction, “module” should be construed to mean “a table or, one or more columns of/within one or more tables.” *Id.* Patent Owner does not cite to any definition of “module” in the ’855 patent Specification or any disclosure that would limit a “module” to a table or a column of a table.

Patent Owner further argues that we should deny institution and “not reach the substance of the dispute” because the Petition does not apply what Petitioner contends to be the proper construction of the term “module.” Prelim. Resp. 17–20. According to Patent Owner, because Petitioner does not apply what it believes is the correct construction, but instead applies a construction Petitioner alleges Patent Owner proposed in the Delaware Lawsuit, Petitioner has failed to comply with the requirements of 37 C.F.R. § 42.104(b)(3). *Id.* at 18–21 (citing *Toyota Motor Corp. v. Blitzsafe Tex., LLC*, IPR2016-00422, Paper 12, 26 (PTAB Jul. 6, 2016)).

In its Preliminary Reply, Petitioner notes that in response to a question during a hearing in the Delaware Lawsuit on a Motion to Dismiss under 35U.S.C. § 101, Patent Owner argued the claim terms recite structure because the claimed module is a table or a portion of a table. Pet. Prelim. Reply 1–2 (citing Ex. 1013, 78). Petitioner also notes that in the Delaware

⁷ Delaware Lawsuit, Transcript of Motions to Dismiss.

Lawsuit, Patent Owner proposed “‘module’ be given its ‘Plain and Ordinary Meaning,’” without defining such plain and ordinary meaning, and that Petitioner proposed that “‘module’ be construed to mean “a table or, one or more columns of/within a single table.” *Id.* at 2 (citing Ex. 1011,⁸ 2).

The premise of Patent Owner’s argument is flawed. Petitioner asserts that, in the context of the ’855 patent, a person of ordinary skill would understand “module” to be a portion of a data repository. Pet. 3. Petitioner also states its understanding that in the Delaware Lawsuit, Patent Owner described a table or a portion of a table as an *example* of a module, i.e., an example of a portion of a data repository. *Id.*; *see* Ex. 1013, 78:13–19 (“This claim has a very specific data dictionary. It’s got particular structures. The modules that are referred to in the claim, an *example* that’s used in the specification is a table or a portion of a table. A table is a structure that people know how to build.”) (emphasis added); *see also* PO Prelim. Sur-reply 1–2 (Patent Owner stating that at the hearing Patent Owner referred to “an example that’s used in the specification is a table or portion of a table”).

Petitioner’s acknowledgement of Patent Owner’s Delaware Lawsuit argument does not change Petitioner’s recognition that the ’855 patent describes a data repository as including modules, such that an ordinarily skilled artisan would have understood a “module” to be a portion of a data repository. The ’855 patent includes an extensive discussion of tables and relationships between tables, but does not define the term “module.” In the ’855 patent, the term “module” appears in the claims; “module” also appears

⁸ Delaware Lawsuit, Joint Claim Construction Chart.

in the “SUMMARY,” in a manner that parrots the language of the claims.
Ex. 1001, 2:1–14.

The term “module” is a classic nonce word that does not connote definite structure. *See, e.g., Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1350 (Fed. Cir. 2015) (“‘Module’ is a well-known nonce word that can operate as a substitute for ‘means’ in the context of [pre-AIA 35 U.S.C.] § 112, para. 6” and “is simply a generic description for software or hardware that performs a specified function” that “does not provide any indication of structure.”). Independent claims 12, 23, and 34 do not use the term “module,” instead reciting limitations similar to those of claims 1–11 using method steps in the case of claims 12 and 23 and express means-plus-function language in the case of claim 34. Ex. 1001, 17:30–53 (claim 12), 18:33–60 (claim 23), 20:10–33 (claim 34).

Claim 34 recites limitations similar to those of claim 1 with express use of “means”-plus-function language, thereby limiting the scope of claim 34 to structures described in the Specification and equivalents thereof. Ex. 1001, 20:10–33; *see* 35 U.S.C. § 112, para. 6 (2012).⁹ Under the

⁹ The method steps of claims 12 and 23 may also be construed under 35 U.S.C. § 112 paragraph 6 as means-plus-function limitations. *See O.I. Corp. v. Tekmar Co.*, 115 F.3d 1576, 1583 (Fed. Cir. 1997) (recognizing that although “[t]he mere fact that a method claim is drafted with language parallel to an apparatus claim with means-plus-function language does not mean that the method claim should be subject to an analysis under § 112, paragraph 6[,] . . . each limitation of each claim must be independently reviewed to determine if it is subject to the requirements of § 112, paragraph 6”). Nonetheless, we need not reach the question of whether the method claims are subject to § 112 paragraph 6 given that claim 34 is a means-plus-function claim and is sufficient to support our claim construction.

doctrine of claim differentiation, we ascribe a broader meaning to the term “module” in claims 1–33 because their construction is not so limited.

Petitioner merely recognizes that in the Delaware Lawsuit Patent Owner attempted to avoid a finding that claims 1–33 recite unpatentable subject matter or are indefinite by ascribing structure to the term “module.” Patent Owner does not contend that the structure claimed is a module, i.e., a table or a portion of a table, is an example of a portion of a data repository. Petitioner’s efforts to address the structure Patent Owner ascribes to the term “module” is not inconsistent with Petitioner’s proposed construction.

To the extent that claims 1–33 can be construed as definite, we construe “module” to mean a portion of a data repository, and note that an example of a structure of a module disclosed in the ’855 patent, i.e., a portion of a data repository, is a table or portion of a table.

IX. ANALYSIS

A. *Legal Principles*

“In an [*inter partes* review], the petitioner has the burden from the onset to show with particularity why the patent it challenges is unpatentable.” *Harmonic Inc. v. Avid Tech., Inc.*, 815 F.3d 1356, 1363 (Fed. Cir. 2016) (citing 35 U.S.C. § 312(a)(3) (requiring *inter partes* review petitions to identify “with particularity . . . the evidence that supports the grounds for the challenge to each claim”)). This burden of persuasion never shifts to Patent Owner. *See Dynamic Drinkware, LLC v. Nat’l Graphics, Inc.*, 800 F.3d 1375, 1378 (Fed. Cir. 2015) (discussing the burden of proof in *inter partes* review).

The question of obviousness is resolved on the basis of underlying factual determinations including: (1) the scope and content of the prior art; (2) any differences between the claimed subject matter and the prior art;

(3) the level of ordinary skill in the art; and (4) objective evidence of obviousness or nonobviousness. *Graham v. John Deere Co.*, 383 U.S. 1, 17–18 (1966). Additionally, the obviousness inquiry typically requires an analysis of “whether there was an apparent reason to combine the known elements in the fashion claimed by the patent at issue.” *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007) (citing *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006) (requiring “articulated reasoning with some rational underpinning to support the legal conclusion of obviousness”)); see *In re Warsaw Orthopedic, Inc.*, 832 F.3d 1327, 1333 (Fed. Cir. 2016) (citing *DyStar Textilfarben GmbH & Co. Deutschland KG v. C. H. Patrick Co.*, 464 F.3d 1356, 1360 (Fed. Cir. 2006)).

An obviousness analysis “need not seek out precise teachings directed to the specific subject matter of the challenged claim, for a court can take account of the inferences and creative steps that a person of ordinary skill in the art would employ.” *KSR*, 550 U.S. at 418; accord *In re Translogic Tech., Inc.*, 504 F.3d 1249, 1259 (Fed. Cir. 2007). Petitioner cannot satisfy its burden of proving obviousness by employing “mere conclusory statements.” *In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016). Instead, Petitioner must articulate a reason why a person of ordinary skill in the art would have combined the prior art references. *In re NuVasive*, 842 F.3d 1376, 1382 (Fed. Cir. 2016).

A reason to combine or modify the prior art may be found explicitly or implicitly in market forces; design incentives; the “interrelated teachings of multiple patents”; “any need or problem known in the field of endeavor at the time of invention and addressed by the patent”; and the background knowledge, creativity, and common sense of the person of ordinary skill.

Perfect Web Techs., Inc. v. InfoUSA, Inc., 587 F.3d 1324, 1328–29 (Fed. Cir. 2009) (quoting *KSR*, 550 U.S. 418–21 (2007)).

In determining whether a claim is obvious in light of the prior art, we consider any relevant objective evidence, when in evidence, of obviousness or non-obviousness. *See Graham*, 383 U.S. at 17. At this stage of the proceeding Patent Owner does not present evidence of such objective considerations

We analyze the asserted grounds of unpatentability in accordance with these principles to determine whether Petitioner has met its burden to establish a reasonable likelihood of success at trial.

B. Claims 1, 3–12, 14–23, 25–33 as Obvious Over Althoff and Surjanto

Petitioner alleges claims 1, 3–12, 14–23, 25–33 would have been obvious over Althoff and Surjanto. Pet. 8–55. Petitioner relies on the White Declaration. Pet. 8 (citing Ex. 1003 ¶¶ 58–158).

1. Althoff (Ex. 1006)

Althoff relates to modeling of object-oriented database structures, translation to relational database structures, and dynamic searches of the same. Ex. 1006, code (57), 1:16–18, 3:12–14. By way of background, Althoff discloses that it is known in software systems to associate data items and data types to allow elements of a system to rely on the characteristics of the data type and relationships between one data type and other data types. Ex. 1006, 1:20–28. Althoff further discloses that users can define a class, e.g., telephone number, to allow elements of the system to store, manipulate, or retrieve telephone numbers as though they were fundamental pieces of information. *Id.* at 1:30–35. Althoff explains that it has been found that the technique of defining classes of software objects and manipulating those objects also has been useful for database applications. *Id.* at 1:41–43. In

particular, according to Althoff, in object-oriented database applications, a user defines classes of software objects, and properties of those classes; and populates a database with data items that are instances of those objects. *Id.* at 1:43–47. For example, Althoff discloses the class “telephone numbers” might be used to store telephone numbers for persons, companies, and computer networks. *Id.* at 1:47–50). Althoff explains that it has been found that object-oriented database management provides advantages such as rapid application and database development and relative software reliability. *Id.* at 1:50–53. Notwithstanding such advantages, however, Althoff further discloses that a problem has arisen in the art insofar as “many database systems have been designed to operate with another, different database technique, known as ‘relational database’ management.” *Id.* at 1:54–57.

In a relational database, Althoff explains,

[T]he database comprises a set of data tables which represent the relationships between data items. Each table comprises a set of records, one record for each data item; each record comprises a set of columns or fields, one column for each data value associated with the data item. (For example, each record in a table of persons might have a column defined for that person's telephone number; Similarly, each record in a table of companies might have a column defined for that company's telephone number.)

Ex. 1006, 1:58–66. According to Althoff, relational databases have been found to be successful and efficient in managing large databases, and accordingly, there has been substantial investment in relational database management systems and the applications and computer programs that support them. *Id.* at 1:66–2:4. Therefore, Althoff discloses, it would be advantageous to obtain the advantages of object-oriented database management while retaining the efficiency of and installed investment in

relational database management systems, but tools for creating, implementing, and manipulating object-oriented databases are generally inconsistent with the tools for creating, implementing, and manipulating relational databases. *Id.* at 2:5–8. Althoff thus seeks to provide a method of compiling and translating between object-oriented and relational database structures that would obtain the advantages of the former while retaining the ability to use the then-existing relational database management systems. *Id.* at 3:4–9.

To achieve the stated goal, Althoff provides a method and system that allow a user to create, edit, and manipulate an object database, objects for that object database, and queries to be applied to the object database, where the method and system dynamically translates the object database, the objects for the object database, and queries to be applied to the object database into a set of relational database structures, data for those relational database structures, and queries to be applied to those relational database structures, respectively. Ex. 1006, 3:12–25. According to Althoff, the method and system “include[] a meta-model of the user’s object database, which is itself an object database, and which has itself been translated into a set of relational database structures for manipulation by a relational database engine.” *Id.* at 3:26–30. The meta-model comprises a set of classes, objects, and relationships between classes that model the classes and relationships between classes of the system. *Id.* at 3:30–32. “Each of these classes may comprise a set of searchable properties, and each of these relationships may comprise an inheritance relationship (between a base class and a derived class) or a data-model relationship (such as a one-to-one, one-to-many, or many-to-many relationship).” *Id.* at 3:31–37. Further, “[t]he data model of the user’s object database is modeled by actual objects in the meta-model,

and editing or manipulating the user's object database is modeled by creating, modifying, or deleting objects in the meta-model." *Id.* at 3:37–41. "The meta-model also models itself, in the same manner as it models the user's object database, and may be manipulated in the same manner as the user's object database." *Id.* at 3:41–44.

In a preferred embodiment, each class in the user's object database is modeled by a relational database table, each searchable property of that class is modeled by a column in that table, and each object of that class is modeled by a row in that table and corresponding rows in tables representing base classes for that object. Each such table comprises an "object-ID" column generated by the system, comprising a unique ID for each object in the system. Each relationship between two objects is modeled by providing a pointer from the first object to the second object; the pointer comprises a column (generated by the system to model the relationship) in the first object's table with the object-ID of the second object. These relational database structures are created, modified, or deleted dynamically or incrementally in response to user commands to create, edit, and manipulate the user's object database.

Id. at 3:44–60. Thus, each class may comprise a set of searchable properties and the relationship may be an inheritance relationship (between a base class and a derived class) or a data-model (e.g., a one-to-one, many-to-one, or many-to-many relationship). *Id.*; *see also id.* at 7:21–25 (describing the meta-model as "comprising an object data base having as its objects the classes 101 in the user's object database 100, searchable properties 102 of those classes, and relationships 103 between those classes").

Metamodel 200 comprises classes shown in table 3–1. Ex. 1006, 10:41–11:10. Each class of the user's object database 100 is modeled by an object of class Class 301. Ex. 1006, 11:13–14. Each object of class Class 301 includes an object ID that is unique for the object, a class label (a name

to be displayed for the class), a class DB name (to be used by the system for generating SQL commands and accessing relational structures corresponding to this class 101), properties 102, such as Class Group, Class Type, and Is System Specific that maintain information about the class 101 in system 200, as well as information about other properties. *Id.* at 11:12–12:23.

2. Surjanto (Ex. 1008)

Surjanto discloses an “object-relational database management system” that “improves content management of XML documents” by “exploiting their structural information.” Ex. 1008, Abstract. In particular, Surjanto discloses that “[w]ith the advent of object-relational database management systems (ORDBMSs) [25], developers can extend the database system.” *Id.* at 9. Surjanto “develop[ed] XCoP [XML Content Repository] as part of an integrated Web database framework . . . built . . . using the extensibility features of an ORDBMS.” *Id.* at 9–10. Surjanto provides benefits such as content reuse, exploitation of structural information, content consistency enforcement, etc. *Id.* at 10.

3. Petitioner’s Contentions Concerning Reasons to Combine Althoff and Surjanto

Petitioner contends that a person of ordinary skill would have had reason to combine the teachings of Althoff and Surjanto to

achieve the well-known benefits of Surjanto’s extensions, which were specifically designed to extend OO databases like Althoff’s to provide the known benefits of XML document support, such as “allow[ing] users to reuse and process textual portions of document contents,” “enabl[ing] collaborative development of documents” and in general “offer[ing] comprehensive content management functionality.”

Pet. 8–9 (citing Ex. 1008, 9; Ex. 1003 ¶ 61).

Petitioner further contends a person of ordinary skill in the art “would be motivated to modify Althoff to include Surjanto’s XCoP extensions because of the well-known benefits of XML document support.” Pet. 9 (citing Ex. 1003 ¶¶ 158–164). Petitioner further identifies Surjanto as disclosing that XCoP allows “users to reuse and process textual portions of document contents.” *Id.*

Patent Owner does not respond to Petitioner’s contention as to combining Althoff and Surjanto. We agree that Petitioner has demonstrated sufficient reasons for combining Althoff and Surjanto.

4. Obviousness of Claim 1

We have reviewed Petitioner’s arguments and evidence regarding obviousness of claim 1. Pet. 8–29; Ex. 1003 ¶¶ 58–118. We have also reviewed relevant portions of the Patent Owner’s Preliminary Response. Prelim. Resp. 21–34.

a) Preamble 1[pre]

The preamble of claim 1 recites, “[a] computerized system for manipulating data sets.” Pet. 12. Neither party contends that the preamble of claim 1 is limiting. Nevertheless, to the extent the preamble of claim 1 is considered limiting, Petitioner cites Althoff’s general purpose computer as teaching a device that allows the user to create, edit, and manipulate the user’s object database. Pet. 12–13 (citing Ex. 1006, 3:15–18). Petitioner further states that an ordinarily skilled artisan would understand that Althoff’s object database includes data sets. *Id.* at 14 (citing Ex. 1003 ¶¶ 70–71). Patent Owner does not respond to Petitioner’s contention concerning the preamble of claim 1. We agree that the Petitioner has demonstrated Althoff discloses the preamble of claim 1.

b) Limitation 1[a]

Petitioner identifies as limitation 1[a] the recitation “a hardware-based processor.” Pet. 13. Noting that Althoff discloses using a general purpose computer, or set of general purpose computers, Petitioner asserts that an ordinarily skilled artisan would understand that such devices include a hardware-based CPU, i.e., a hardware-based processor as claimed. *Id.* (citing Ex. 1006, 5:21–31; Ex. 1003 ¶ 73). Patent Owner does not respond to Petitioner’s contention as to limitation 1[a]. We agree that Petitioner has demonstrated that Althoff discloses claim limitation 1[a].

c) Limitation 1[b]

Limitation 1[b] recites “a data repository that contains the data sets and one or more layers of metadata of the data in the data sets.” Pet. 13. Petitioner asserts that a person of ordinary skill would have understood Althoff’s relational database to be a specific implementation of the data repository as claimed. *Id.* (citing Ex. 1006, 6:42–55; Ex. 1003 ¶¶ 74–80).

In support of its contention that Althoff discloses a data repository that is adapted to store data contained in the data sets and one or more layers of metadata included in the data sets, Petitioner cites Althoff’s teaching of the “user’s object database.” Pet. 13 (citing Ex. 1006, 3:25–29). According to Petitioner, an ordinarily skilled artisan would understand that an object database contains data sets in the form of objects and the data associated with those objects. *Id.* (citing Ex. 1003 ¶¶ 74–75, 77). Petitioner further cites Althoff as disclosing a meta-model of the user’s object database. *Id.* Petitioner notes that Althoff’s meta-model itself is an object database comprising a set of classes, objects and relationships between classes that model the classes and relationship between classes of the system. *Id.* at 14 (citing Ex. 1006, 3:25–32). According to Petitioner, a person of ordinary

skill would understand that Althoff's meta-model of the user's object database would contain one or more layers of metadata sets because the meta-model includes data about the data sets. *Id.* (citing Ex. 1003 ¶¶ 79–80).

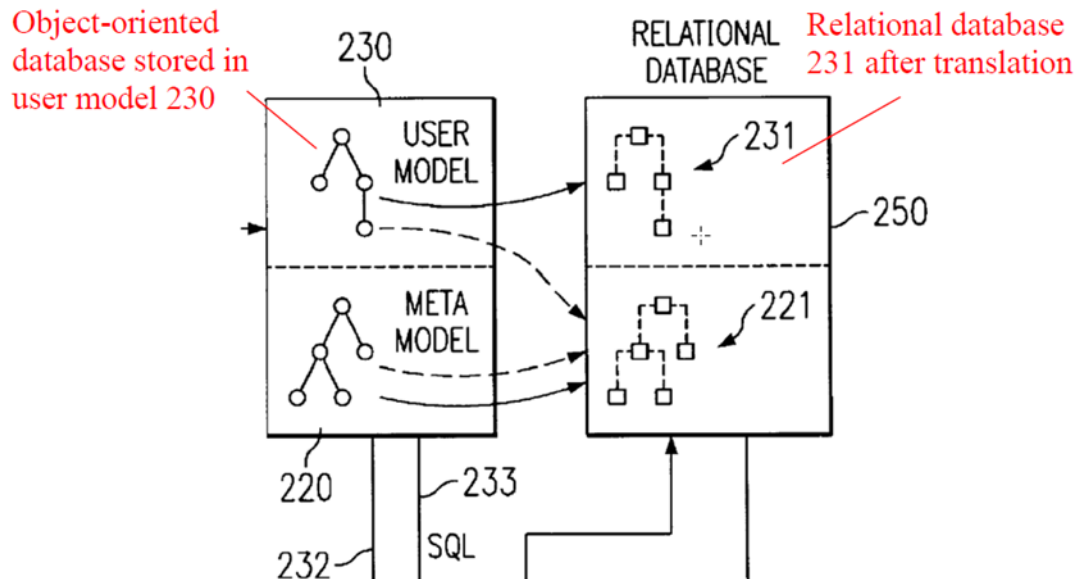
Patent Owner does not respond explicitly to Petitioner's contentions concerning claim limitation 1[b]. We agree that Petitioner has demonstrated that Althoff discloses limitation 1 [b].

d) Limitation 1[c]

Limitation 1[c] recites “a class module configured to define at least one class of elements and store the class.” Pet. 14. According to Petitioner, “the *elements* stored in the *element module* are instanced objects of a class.” *Id.* (citing Ex. 1006, 3:45–46 (“each class . . . is modeled by a relational database table . . . and each object of that class is modeled by a row in that table.”); Ex. 1003 ¶¶ 81–84). Petitioner notes that Althoff teaches that “in the meta-model 220, *each class of the user's object database 100 is modeled* by an object of class Class 301” that can be defined by a user. *Id.* (citing Ex. 1006, 11:13–14, 1:31–32).

According to Petitioner, Althoff discloses storing the class by teaching that in the “‘class Class 301’ table, where ‘each class . . . is modeled by an object [*i.e.* row] of class Class 301’ and ‘each object of class Class 301 comprises information in table 3-2.’” Pet. 15 (citing Ex. 1006, 11:12–16; Ex. 1003 ¶ 83). Petitioner alleges that “Althoff teaches that these class modules are defined and stored in a “class Class 301” table (*i.e.*, class module).” *Id.* at 14 (citing Ex. 1003 ¶ 83). Petitioner alleges the information stored includes the “name to be displayed for the class” and a “description of the class.” *Id.* at 15 (citing Ex. 1006, 11:18–48).

Patent Owner argues that numerous arguments in the Petition, including those relating to limitation 1[c], “fail because the Petition ignores the distinction between object database 100 in user model 230 and relational database 231.” Prelim. Resp. 21–22. This argument is supported by Patent Owner’s assertion that “Althoff discloses an ‘object database 100,’ depicted in Figure 1, that is stored in ‘user model 230,’ and that is **translated into** ‘relational database 231,’ depicted in Figure 8.” *Id.* at 21 (citing Ex. 1006, 2, 8, 8:18–28, 23:62, 24:31, Figs 1, 8. The argument is illustrated in Patent Owner’s annotation of Figure 2 of Althoff.



Patent Owner’s Annotated Figure 2 of Althoff.

Prelim. Resp. 21. Thus, Patent Owner argues “[a]ccordingly, Althoff proposes its translator between **two distinct databases**, and provides an exemplary object database in Figure 1 and an exemplary counterpart relational database in Figure 8.” *Id.* at 22 (citing Ex. 1006, 3:4–7, 1:14–18).

Patent Owner points out that claim 1, as recited in limitation 1[c], requires the data repository to contain a “class module configured to define

at least one class of elements and store the class.” Prelim. Resp. 22 (citing Ex. 1001, 16:28–29). Patent Owner contends the Petitioner asserts that “Althoff disclose the data repository as a ‘relational database . . . implemented using a relational database engine.” *Id.* at 22–23 (citing Pet. 13 (showing for limitation 1[b])). Patent Owner continues by arguing that a person of ordinary skill in the art “would understand that a relational database is a specific implementation of a ***data repository***.” *Id.* at 23 (citing Pet. 13). Patent Owner concludes that “Petitioner needed to identify a module (using its construction of module) in Althoff’s relational database that is configured to define at least one class,” but failed to do so. *Id.*

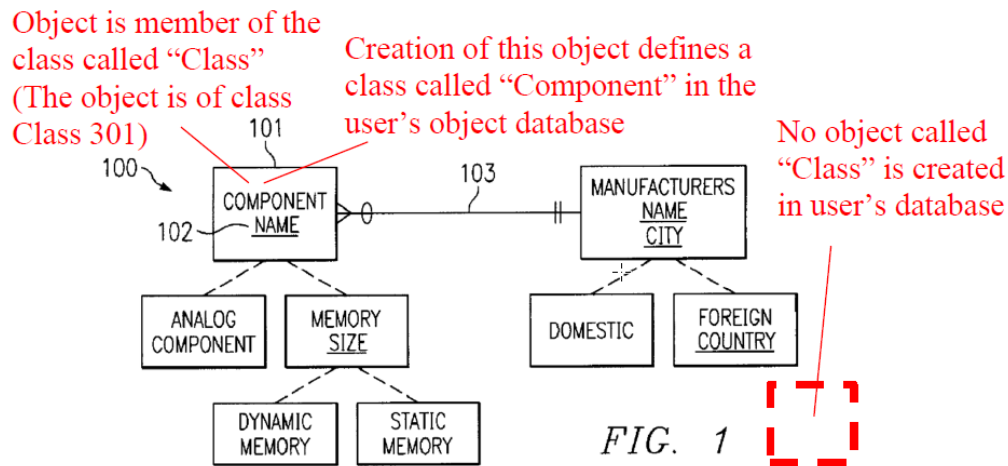
According to Patent Owner, Class 301 in Althoff’s meta-model is a data type used by the object database to identify the classes in the user’s database. Prelim. Resp. 23. Referencing Althoff’s Figure 1, Patent Owner states that Althoff’s system

defines classes by creating objects that are identified as having the type “Class.” Thus, each of the classes in Figure 1 is defined in Althoff’s object database by creating objects of class Class (that is, objects that are members of the class called “Class”).

Id. at 26 (citing Ex. 1006, 10:41–46).

Patent Owner argues “‘class Class 301’ is a class (a data type) defined in Althoff’s meta-model,” which is in the object database. Prelim. Resp. 26 (citing Ex. 1006, 10:41–46). Patent Owner argues “[e]ach object is member of the class Class 301” but there is “no object called ‘Class’ in the user’s object database.” *Id.* at 26–27 (citing Ex. 1006, 11:12–14). Patent Owner notes that Althoff’s generates eight classes in the user “object database: (1) Component, (2) Analog Component, (3) Memory, (4) Dynamic Memory, (5) Static Memory, (6) Manufacturer, (7) Domestic [manufacturer], and (8)

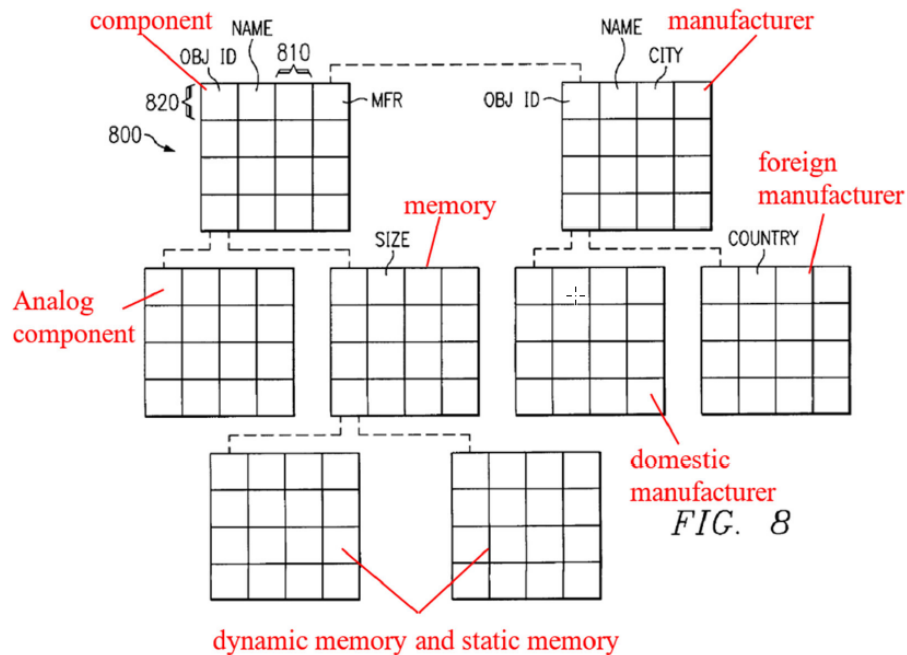
Foreign [manufacturer]).” *Id.* at 27 (citing Ex. 1006, 5:33–6:29, Fig. 1).
Patent Owner’s annotation of Figure 1 is reproduced below.



Patent Owner’s Annotated Figure 1 of Althoff.

Prelim. Resp. 27. Annotated Figure 1 of Althoff illustrates Patent Owner’s argument that there is no object called “Class” in the user’s object database. *Id.* at 26–27. As noted above, Althoff’s class Class 301 is a class or data type in Althoff’s meta model identified as classes in the user’s object database. *Id.* at 26 (citing Ex. 1006, 10:41–46).

Patent Owner argues that “class Class 301” is not translated from the object database to the relational database shown in Figure 2 above. Prelim. Resp. 27. This argument is illustrated by Patent Owner’s annotation of Figure 8 of Althoff, which is reproduced below.



Patent Owner's Annotated Figure 1 of Althoff.

Prelim. Resp. 29 (citing Ex. 1006, 4:58–59, 5:8–10, Fig. 8). Figure 1 is translated into Figure 8. *Id.* at 21 (citing Ex. 1006, 8:18–28, 23:62–24:31, Figs. 1, 2, 8). As discussed above, Patent Owner argues annotated Figure 8 of Althoff supports its argument that “there is no class in the user’s object database called ‘Class,’ the translator will not select this non-existent class for translation into relational database tables.” *Id.*

We agree with Patent Owner that, contrary to Petitioner’s assertion, Althoff does not teach the claimed class module, i.e., a module (or table) configured to define at least one class of the plurality of elements (Althoff’s objects 101) and store that class. The claimed “data repository” is the relational database. *See* Pet. 13. However, Class 301 is part of the object database and is not translated into the relational database. As Patent Owner argues, Althoff teaches that the system “selects a class 101 in the user’s object database 100,” not the relational database. *See* Prelim. Resp. 27

(citing Ex. 1006, 23:6–7, 5:39–40 (“An example user’s object database 100 [in FIG. 1] comprises a set of classes 101.”)).

Patent Owner’s contention that there would be no reason to generate a class Class 301 table as Petitioner asserts is also persuasive. *See* Prelim. Resp. 29–30. As Patent Owner notes, by creating separate tables for each class, e.g., a Component table and a Manufacturer table, Althoff preserves the class information. *Id.* Patent Owner also argues that Petitioner’s assertions concerning a class Class 301 table fail because “if objects were already stored in a table in the object database, there would be no need for Althoff to build a translator to translate objects into tables.” *Id.* at 31.

For at least the foregoing reasons, we conclude that Petitioner has not shown adequately at this stage of the proceeding that Althoff teaches or suggests limitation 1[c]. Moreover, Petitioner does not allege that Surjanto addresses the above-identified deficiencies of Althoff with respect to that limitation. Thus, we determine that Petitioner has not demonstrated that there is a reasonable likelihood it will succeed in its challenge to claim 1. We need not address the remaining limitations of claim 1 or Patent Owner’s arguments as to why those limitations have not been sufficiently shown. *See* Prelim. Resp. 34–40.

5. Independent Claims 12 and 23

Claim 12 is drawn to a method. Claim 23 is drawn to a non-transitory computer medium. Petitioner relies for claims 12 and 23 on its arguments with respect to claim 1, merely citing for the limitations of claims 12 and 23 to corresponding limitations of claim 1 and arguing that claims 12 and 23 are unpatentable for the same reasons as alleged for claim 1. Pet. 45–47. For the reasons discussed above with respect to claim 1, we determine that

Petitioner also has not demonstrated a reasonable likelihood it would succeed in its challenge to claims 12 and 23.

6. Claims 2–11, 13–22, and 24–33

Claims 2–11 depend from claim 1. Claims 13–22 depend from independent claim 12. Claims 24–33 depend from independent claim 23. As discussed above, Petitioner has not identified disclosure in Althoff and Moore that teaches the subject matter of the independent claims discussed above that we found lacking in Althoff. Therefore, we determine that Petitioner has not demonstrated a reasonable likelihood it would succeed in its challenge to claims 2–11, 13–22, and 24–33.

C. Petitioner’s Contentions That Claims 2, 13, and 24 Would Have Been Obvious over Althoff, Surjanto, and Nye

Petitioner alleges claims 2, 13, and 24 would have been obvious over Althoff, Surjanto, and Nye. Pet. 47–55. Petitioner relies on the White Declaration. Pet. 47 (citing Ex. 1003 ¶¶ 157–177).

Petitioner relies on Nye as disclosing an event model that allows a developer to add complex dependency logic to an existing database without having to modify the underlying structure of the database. Pet. 7, 47–51. According to Petitioner, a person of ordinary skill would have understood that the integration of state machines in the object-oriented model taught by Althoff would provide benefits, e.g., by allowing a developer to describe different modes or operation of the system, and to make the system more robust. *Id.*

Claim 2 depends from independent claim 1. Claim 13 depends from independent claim 12. Claim 24 depends from independent claim 23. Petitioner has not identified disclosure in Nye that teaches the subject matter of the independent claims lacking in Althoff. Therefore, we determine that

Petitioner has not demonstrated a reasonable likelihood it will succeed in its challenge to claims 2, 13, and 24 as unpatentable over Althoff, Surjanto, and Nye

*D. Petitioner’s Contentions that Claims 4, 67, 15, 1718, 26, 28, and 29
Would Have Been Obvious Over Althoff, Surjanto, and Moore*

Petitioner alleges claims 4, 6–7, 15, 17, 18, 26, 28, and 29 would have been obvious over Althoff, Surjanto, and Nye. Pet. 56–68. Petitioner relies on the White Declaration. Pet. 56 (citing Ex. 1003 ¶¶ 179–188).

Petitioner relies on Moore as disclosing that a property is “stored in the [relational] database and [is] represent[ed] as a container that is like a folder.” Pet. 8 (citing Ex. 1007, 3:39–41) (first set of brackets in the original). In addition, Moore is cited for teaching “the well-known benefit of giving end-users a familiar, folder-based user interface to navigate hierarchical database objects, such as those taught in Althoff (and Surjanto).” *Id.* at 57 (citing Ex. 1003 ¶ 181).

Claims 4, 6, and 7 depend from independent claim 1. Claims 15, 17, and 18 depend from independent claim 12. Claims 26, 28, and 29 depend from independent claim 23. Petitioner has not identified disclosure in Moore that teaches the subject matter of the independent claims lacking in Althoff. Therefore, we determine that Petitioner has not demonstrated a reasonable likelihood it would succeed in its challenge to claims 4, 6, 7, 15, 17, 18, 26, 28, and 29 as unpatentable over Althoff, Surjanto, and Moore.

X. CONCLUSION

For the reasons discussed above, we determine that Petitioner has not demonstrated a reasonable likelihood that it would succeed on any of its challenges to patentability.

XI. ORDER

In consideration of the foregoing, it is hereby:
ORDERED that the Petition is DENIED.

IPR2023-01073
Patent 8,224,855 B2

FOR PETITIONER:

James M. Glass
John T. McKee
Quincy Lu
Joshua Parzivand
QUINN EMANUEL URQUHART & SULLIVAN, LLP
jimglass@quinnemanuel.com
johnmckee@quinnemanuel.com
quincylu@quinnemanuel.com
joshuaparzivand@quinnemanuel.com

FOR PATENT OWNER:

Steven Katz
Andrew Pearson
Jason Wolff
FISH & RICHARDSON P.C.
katz@fr.com
pearson@fr.com
wolff@fr.com