



UNITED STATES PATENT AND TRADEMARK OFFICE

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

Table with 5 columns: APPLICATION NO., FILING DATE, FIRST NAMED INVENTOR, ATTORNEY DOCKET NO., CONFIRMATION NO.
12/766,416 04/23/2010 Richard Gary McDaniel 2009P07377 US01 1385

45113 7590 11/28/2016
Siemens Corporation
Intellectual Property Department
3501 Quadrangle Blvd Ste 230
Orlando, FL 32817

EXAMINER

OCHOA, JUAN CARLOS

ART UNIT PAPER NUMBER

2123

NOTIFICATION DATE DELIVERY MODE

11/28/2016

ELECTRONIC

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

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

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

IPDadmin.us@siemens.com

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE PATENT TRIAL AND APPEAL BOARD

---

*Ex parte* RICHARD GARY McDANIEL<sup>1</sup>

---

Appeal 2015-004531  
Application 12/766,416  
Technology Center 2100

---

Before KRISTEN L. DROESCH, JOHNNY KUMAR, and  
MATHEW R. CLEMENTS, *Administrative Patent Judges*.

DROESCH, *Administrative Patent Judge*.

DECISION ON APPEAL

STATEMENT OF THE CASE

Appellant seeks review under 35 U.S.C. § 134(a) from the Examiner's Final Rejection of claims 1–21, all of the pending claims in the application.

We have jurisdiction under 35 U.S.C. § 6(b).

We REVERSE.

BACKGROUND

The disclosed invention is directed to systems and methods for use in computer-aided design (“CAD”), manufacturing, engineering, prototype/test,

---

<sup>1</sup> Appellant indicates the real party-in-interest is Siemens Product Lifecycle Management Software Inc. App. Br. 4.

maintenance, modeling, and visualization (“CAD systems”) and product lifecycle management (“PLM”). *See* Spec. ¶ 2.

#### CLAIMED SUBJECT MATTER

Representative claim 1 reproduced from the Claims Appendix of the Appeal Brief, reads as follows:

1. A method for managing behavior of a CAD model, comprising:

receiving a geometric object and corresponding physical object, for the CAD model, in a data processing system, and receiving a corresponding user annotation that specifies a simulation behavior by associating the geometric object with the corresponding physical object, wherein the physical object describes physical qualities and behaviors of the corresponding geometric object, including at least one of movement, collisions, connections, mass, velocity, or force;

executing function code in a behavior object by the data processing system, to determine required references of the behavior object, wherein the required references indicate one or more physical objects that are required for execution of the behavior object;

receiving and storing an assignment of at least one of the required references of the behavior object to the physical object by the data processing system;

storing the geometric object, physical object, and behavior object in the data processing system as associated with the CAD model;

simulating operation of at least part of the CAD model by the data processing system according to the specified simulation behavior, including executing the behavior object by the data processing system to modify a state of the physical object.

## REJECTION ON APPEAL

Claims 1–21 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Grinstein et al. (US 6,714,201 B1; Mar. 30, 2004) (“Grinstein”), and Charles et al. (US 2007/0013709 A1; Jan. 18, 2007) (“Charles”).

## ANALYSIS

We have reviewed the Examiner’s rejection in light of Appellant’s arguments in the Appeal Brief, the Examiner’s Answer, and the arguments in the Reply Brief. We agree with Appellant’s arguments. We highlight and address specific findings and arguments below for emphasis.

The Appellant contests the Examiner’s finding that Grinstein teaches or suggests “executing function code in a behavior object by the data processing system, to determine required references of the behavior object,” as recited in independent claim 1, and recited similarly in independent claims 8 and 15. *See* App. Br. 23–28, 42–44, 56–58; Reply Br. 24–28. The Examiner makes the following findings:

**executing function code** (see "function code" as "The basic functional code of the System is called the Run-time Engine 130 (RTE)" in col. 10, line(s) 24-29) **in a behavior object by the data processing system** (see "motions, behaviors, and boundaries, are each modeled as classes in an object oriented language such as c++. Since behaviors are attributes of motions, once a given class of behavior has been defined, it can be used to control different motions" in col. 5, line 66 to col. 6, line 11; "OpenMotion API uses a class called Behavior" in col. 29, line(s) 7-15), **to determine required references of the behavior object, wherein the required references indicate one or more physical objects that are required for execution of the behavior object** (see "references" as "Expressions" of 'An instance of the motion class, or motion, contains state information for multiple spatial degrees of freedom, such as translation, rotation, and scaling which can be associated with an

object. Each degree of freedom has a characteristic trajectory. Motions may be combined by expressions that specify hierarchical parent-child relationships, expressions that specify blending of simultaneous motions, and expressions that extract attribute information" in col. 17, line 66 to col. 18, line 7; "references" as "Expressions" of 'Expressions, referred to as "algebraic expressions" above, may be regarded as the "cables" that interconnect the elements of the motion model. The programmer uses expressions to link motions, behaviors, and boundaries to each other, and to transmit data to and from application specific code' in col. 18, line(s) 34-46).

Final Act. 4–5; Ans. 19–20. The Examiner further explains:

[s]ince Grinstein’s ‘RTE contains all of the functions, which are accessible through the API 102 as well as some which are accessible only from within the code of the RTE itself,’ it is the Examiner’s position that any one of ordinary skill in the art can recognize Grinstein’s RTE as ‘function code’.”

Ans. 20 (citing Grinstein 10:24–29, Fig. 1).

We agree with Appellant’s argument that Grinstein’s Run-time Engine (“RTE”) is not “function code *in a behavior object*.” See App. Br. 24, 42–43, 56–57; Reply Br. 26, 30. Specifically, the RTE is not “in a behavior object” (i.e., Grinstein’s classes), as claimed, but rather *contains* all “behavior objects.” Grinstein, 10:24–29 (“The Run-time Engine (‘RTE’) *contains* all of the functions, types, and classes which are accessible through the API 102 . . .”) (emphasis added). We also agree with Appellant’s argument that there is no teaching or suggestion in Grinstein “that executing code in a behavior class determines required references (or even “expressions”) for a behavior object, as would be required by the Office Action’s analysis.” App. Br. 28, 44, 58; Reply Br. 28, 30. As applied by the Examiner, the teachings of Charles do not remedy the deficiencies of Grinstein. See Final Act. 6

Appeal 2015-004531  
Application 12/766,416

Accordingly, for these reasons, we are constrained to reverse the Examiner's rejection of claims 1–21 as unpatentable over Grinstein and Charles.

DECISION

We REVERSE the rejection of claims 1–21.

REVERSED