IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent of:	Ezra Eddie Bakhash
U.S. Patent No.:	9,696,868
Issue Date:	July 4, 2017
Appl. Serial No.:	14/614,708
Filing Date:	February 5, 2015
Title:	SYSTEM AND METHOD FOR PROVIDING THREE-DIMENSIONAL GRAPHICAL USER INTERFACE

Mail Stop Patent Board
Patent Trial and Appeal Board
U.S. Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

PETITION FOR *INTER PARTES* REVIEW OF UNITED STATES PATENT NO. 9,696,868 PURSUANT TO 35 U.S.C. §§ 311–319, 37 C.F.R. § 42

TABLE OF CONTENTS

I. INTRODUCTION 1

II. REQUIREMENTS FOR IPR 4
- **A.** Standing 4
- **B.** Challenge and Relief Requested 4

III. OVERVIEW OF THE '868 PATENT 5
- **A.** Brief Description 5

IV. Level of Ordinary Skill in the Art 9

V. CLAIM CONSTRUCTION—37 C.F.R. §42.104(b)(3) 9

VI. THE CHALLENGED CLAIMS ARE UNPATENTABLE 10
- **A.** GROUNDS 1A/2A: Claims 1-5, 9-14, and 18-20 are obvious over Hanggie in view of Anthony or Anthony in view of Hanggie 10
  - 1. Anthony 10
  - 2. Hanggie 14
  - 3. Combination of Anthony and Hanggie 22
  - 4. Combination of Hanggie and Anthony 30
  - 5. Manner in which Grounds 1A and 2A render obvious Claims 1-10 and 13-30 34
- **B.** GROUNDS 1B/2B – Claims 6-8 and 15-17 is obvious over HAC and Matthews 77
  - 1. Matthews 77
  - 2. Combination of Matthews with HAC 79
  - 3. Manner in which HAC and Matthews render claims 6-8 and 15-17 obvious 80
- **C.** GROUNDS 1C/2C – Claim 7 is obvious over HAC, Matthews, and Robertson 90
  - **1.** Robertson 90
  - **2.** Combination of Robertson with HAC-Matthews 91
  - **3.** Manner in which HAC, Matthews, and Robertson render obvious claim 7 92

VII. PTAB DISCRETION SHOULD NOT PRECLUDE INSTITUTION 93
- **A.** 35 U.S.C. §325(d)—*Advanced Bionics* 93
- **B.** 35 U.S.C. §314(a)—*Fintiv* 94
  - 1. Factor 1: Petitioner Requested a Stay 95
  - 2. Factor 2: The Board's Statutory Timeline is More Reliable Than the District Court's 95

3. Factor 3: Petitioner's Diligence and Investment in IPR Favors Institution....96
4. Factor 4: The Petition's Grounds are Materially Different from Any That Might be Raised in Litigation 96
5. Factor 5: Parties in Parallel Proceedings....96
6. Factor 6: The Merits of this Petition Compel Institution....97

VIII. CONCLUSION....98

IX. PAYMENT OF FEES – 37 C.F.R. §42.103 98

X. MANDATORY NOTICES—37 C.F.R §42.8(a)(1) 98
A. Real Party-In-Interest—37 C.F.R. §42.8(b)(1)....98
B. Related Matters—37 C.F.R. §42.8(b)(2)....99
C. Lead And Back-Up Counsel—37 C.F.R. §42.8(b)(3)....99
B. Service Information 101

EXHIBITS

APPLE1001	U.S. Patent No. 9,696,868 to Ezra E. Bakhash ("the '868 Patent")
APPLE1002	Excerpts from the Prosecution History of the '868 Patent ("the Prosecution History")
APPLE1003	Declaration of Dr. Henry Fuchs
APPLE1004	Complaint, *SpaceTime3D, Inc. v Apple Inc.*, 6-22-cv-00149, (W.D. Tex.), Feb. 10, 2022
APPLE1005	Apple Opening Claim Construction Brief, *SpaceTime3D, Inc. v Apple Inc.*, 6-22-cv-00149, (W.D. Tex.), Sept. 1, 2022
APPLE1006	US Patent Pub. No. 2005/0088447 ("Hanggie")
APPLE1007	US Patent Pub. No. 2005/0091596 ("Anthony")
APPLE1008	US Patent Pub. No. 2006/0107229 ("Matthews")
APPLE1009	US Patent No. 6,414,677 ("Robertson")
APPLE1010	Dictionary of Electrical & Computer Engineering, McGraw-Hill, 2004.
APPLE1011	US Patent Pub. No. 2012/0053926 ("Satpute")
APPLE1012	US Patent No. 5,664,896 ("Blumberg")
APPLE1013	US Patent No. 7,536,676 ("Baker")
APPLE1014-APPLE1019	RESERVED
APPLE1020	U.S. District Courts–Combined Civil and Criminal Federal Court Management Statistics (June 30, 2022) | United States Courts (uscourts.gov),

	https://www.uscourts.gov/sites/default/files/fcms_na_distprofile0630.2022_0.pdf (last visited December 9, 2022)
APPLE1021	Memorandum, Interim Procedure for Discretionary Denials in AIA Post-Grant Proceedings with Parallel District Court Litigation (USPTO June 21, 2022) ("Interim Procedure")
APPLE1022	Order Cancelling *Markman* Hearing, *SpaceTime3D, Inc. v Apple Inc.*, 6-22-cv-00149, (W.D. Tex.)
APPLE1023	Apple's Opposed Motion to Stay Pending Transfer, *SpaceTime3D, Inc. v. Apple Inc*., Case No.:6:22-cv-00149 (WDTX)
APPLE1024	Scheduling Order, *SpaceTime3D, Inc. v. Apple Inc*., Case No.:6:22-cv-00149 (WDTX)
APPLE1025	Order Resetting *Markman* Hearing, *SpaceTime3D, Inc. v. Apple Inc*., Case No.:6:22-cv-00149 (WDTX)
APPLE1026	Stipulation by Apple, *SpaceTime3D, Inc. v. Apple Inc*., Case No.:6:22-cv-00149 (WDTX)
LGE1027	Declaration of Service of Summons, *SpaceTime3D, Inc. v. LG Electronics U.S.A., Inc., et al.*, Case No 2:22-cv-00049-RWS (EDTX)
LGE1028	Docket Control Order, *SpaceTime3D, Inc. v. LG Electronics U.S.A., Inc., et al.*, Case No. 2:22-cv-00049-RWS (EDTX)
LGE1029	Email Between Counsel Regarding Electronic Service

LIST OF CHALLENGED CLAIMS

Claim 1	
[1pre]	A method for using a two-dimensional (2D) space to selectively interact with a plurality of applications open on a device and a three-dimensional (3D) immersive space to switch between said plurality of application, said device having a fixed resolution display, comprising:
[1a]	receiving a plurality of inputs from a user, said plurality of inputs comprising at least first, second, and third inputs;
[1b]	opening said plurality of applications in response to said plurality of inputs, said plurality of applications comprising at least first, second, and third applications, wherein for each one of said plurality of applications (i) an object is generated having application-specific data, (ii) said object is displayed in said 2D space on said fixed resolution display, and (iii) said user is allowed to modify at least a portion of said application-specific data by interacting with said object;

[1c]	allowing a user to switch between said plurality of application, comprising;
[1c-1]	generating at least a plurality of images, said plurality of images comprising at least first, second, and third images, wherein said first image is an image of at least a portion of a first object generated by said first application and having first application-specific data, said second image is an image of at least a portion of a second object generated by said second application and having second application-specific data, and said third image is an image of at least a portion of a third object generated by said third application and having third application-specific data;

[1c-2]	replacing all objects corresponding to said plurality of applications that are visible in said 2D space with said plurality of images, said plurality of images being displayed in said 3D immersive space and in an order based on a last time that said user one of (i) opened said first application and interacted with said first object, (ii) opened said second application and interacted with said second object, and (iii) opened said third application and interacted with said third object, wherein a first one in said order is displayed in a foreground of said 3D immersive space, a second one in said order is displayed in a background of said 3D immersive space behind at least said first one in said order, and a third one in said order is displayed in said background of said 3D immersive space behind at least said second one in said order;
[1c-3]	allowing said user to move said plurality of images, wherein (i) movement of one of said plurality of images results in movement of all of said plurality of images, and

[1c-4]	(ii) continued movement in one direction of one of said plurality of images results in a perception to said user that said one of said plurality of images is moved off of said fixed resolution display; and
[1c-5]	allowing said user to delete at least one of said plurality of images from said 3D immersive space, wherein deletion of said second one in said order results in said third one in said order being moved to a location in said 3D immersive space where said second one in said order was located prior to said deletion; and
[1d]	allowing said user to interact with one of said first, second, and third applications, comprising:
[1d-1]	receiving a selection from said user of one of said plurality of images corresponding to one of said plurality of applications;
[1d-2]	replacing said plurality of images within said 3D immersive space with one of said first, second, and third objects corresponding to said one of said plurality of applications within said 2D space in response to said selection;

[1d-3]	receiving at least one interaction by said user with said one of said first, second, and third objects within said 2D space; and
[1d-4]	modifying said one of said first, second, and third application-specific data in response to said at least one interaction.
Claim 2	
[2]	The method of claim 1, wherein said step of displaying said plurality of images on said display device further comprises displaying said plurality of images together with a plurality of icons, wherein each one of said plurality of images has a corresponding one of said plurality of icons.
Claim 3	
[3]	The method of claim 1, wherein said step of displaying said plurality of images on said display device further comprises displaying said plurality of images together with a plurality of descriptions, wherein each one of said plurality of images has a corresponding one of said plurality of descriptions.

Claim 4	
[4]	The method of claim 1, further comprising the step of storing in a database data linking each one of said plurality of images with a corresponding one of said plurality of applications and a corresponding one of said order in which said plurality of images are displayed on said display device.
Claim 5	
[5]	The method of claim 1, wherein a reduction in size of an object or portion thereof is used to depict said object or portion thereof as being behind at least one other object or other portion of said object.
Claim 6	
[6a]	The method of claim 1, further comprising the steps of: displaying a fourth image on said display device, said fourth image being an image of a desktop having at least one other application; and

[6b]	allowing said user to open said other application by allowing said user to select said fourth image, replacing said fourth image with said desktop in response to said selection, receiving an interaction from said user with said other application, and opening said other application in response to said interaction.
Claim 7	
[7]	The method of claim 1, wherein each one of said plurality of applications comprises a web browser application, and each one of said first, second, and third objects comprises a web browser window.
Claim 8	
[8a]	The method of claim 7, further comprising the steps of: displaying at least fourth and fifth images in a second 3D immersive space, wherein said fourth image is an image of said plurality of images within said 3D immersive space and said fifth image is an image of at least a portion of a fourth object generated by a fourth application previously opened by said user; and

[8b]	allowing said user to interact with said one of said first, second, and third applications by selecting said fourth image, selecting said one of said plurality of images corresponding to said one of said plurality of applications, and interacting with said one of said first, second, and third objects corresponding to said one of said plurality of applications.
Claim 9	
[9]	The method of claim 1, wherein said step of replacing said plurality of images with one of said first, second, and third objects, further comprises enlarging a size of at least said one of said plurality of images before replacing said one of said plurality of images with said one of said first, second, and third objects, thereby simulating movement of said one of said plurality of images in a z-axis of said display device.

Claim 10	
[10pre]	A system for using a two-dimensional (2D) space to selectively interact with at least one of a plurality of applications open on a computing device and a three-dimensional (3D) space to switch between said plurality of application, comprising:
[10a]	a display device;
[10b]	at least one input device;
[10c]	at least one processor operatively coupled to said display device and said at least one input device; and
[10d]	a memory device comprising executable code for:
[10d-1]	receiving a plurality of inputs from a user, said plurality of inputs comprising at least first, second, and third inputs;

[10d-2]	opening said plurality of applications in response to said plurality of inputs, said plurality of applications comprising at least first, second, and third applications, wherein for each one of said plurality of applications (i) an object is generated having application-specific data, (ii) said object is displayed in said 2D space on said fixed resolution display, and (iii) said user is allowed to modify at least a portion of said application-specific data by interacting with said object;
[10d-3]	allowing a user to switch between said plurality of application, comprising;

[10d-3i]	generating at least a plurality of images, said plurality of images comprising at least first, second, and third images, wherein said first image is an image of at least a portion of a first object generated by said first application and having first application-specific data, said second image is an image of at least a portion of a second object generated by said second application and having second application-specific data, and said third image is an image of at least a portion of a third object generated by said third application and having third application-specific data;

[10d-3ii]	replacing said object corresponding to one of said plurality of applications that is being interacted with by said user in said 2D space with said plurality of images, said plurality of images being displayed in said 3D space and in an order based on a last time that said user one of (i) opened said first application and interacted with said first object, (ii) opened said second application and interacted with said second object, and (iii) opened said third application and interacted with said third object, wherein a first one in said order is displayed in a foreground of said 3D space in front of at least said second one in said order, said second one in said order is displayed in said 3D space behind at least said first one in said order, and a third one in said order is displayed in said 3D space behind at least said second one in said order;
[10d-3iii]	allowing said user to move said plurality of images, wherein (i) movement of one of said plurality of images results in movement of all of said plurality of images, and

[10d-3iv]	(ii) continued movement in one direction of one of said plurality of images results in a perception to said user that said one of said plurality of images is moved off of said fixed resolution display; and
[10d-3v]	allowing said user to delete at least one of said plurality of images from said 3D space, wherein deletion of said second one in said order results in said third one in said order being moved to a location in said 3D space where said second one in said order was located prior to said deletion; and
[10d-4]	allowing said user to interact with one of said first, second, and third applications, comprising:
[10d-4i]	receiving a selection from said user of one of said plurality of images corresponding to one of said plurality of applications;
[10d-4ii]	replacing said plurality of images within said 3D space with one of said first, second, and third objects corresponding to said one of said plurality of applications within said 2D space in response to said selection;

[10d-4iii]	receiving at least one interaction by said user with said one of said first, second, and third objects within said 2D space; and
[10d-4iv]	modifying said one of said first, second, and third application-specific data in response to said at least one interaction.
Claim 11	
[11]	The system of claim 10, wherein said step of displaying said plurality of images on said display device further comprises displaying said plurality of images together with a plurality of icons and a plurality of descriptions, wherein each one of said plurality of images has a corresponding one of said plurality of icons and a corresponding one of said plurality of descriptions.
Claim 12	
[12]	The system of claim 10, further comprising periodically replacing at least one of said plurality of images with at least one other image, said other image showing at least one change to application specific data of a corresponding to one of said plurality of applications while said plurality of images are being displayed in said 3D space.

Claim 13	
[13]	The system of claim 12, further comprising the step of receiving a request from said user to switch from said 2D space to said 3D space, wherein said step of replacing said object in said 2D space with said plurality of images further comprises replacing said object in said 2D space with said plurality of images in response to receiving said request to switch from said 2D space to said 3D space.
Claim 14	
[14]	The system of claim 10, wherein a reduction in size of an object or portion thereof is used to depict said object or portion thereof as being behind at least one other object or other portion of said object.
Claim 15	
[15a]	The system of claim 10, wherein said executable code is further configured to: display a fourth image on said display device in said 3D space, said fourth image being an image of a desktop having at least one other application; and

[15b]	allow said user to open said other application by allowing said user to select said fourth image, replacing said fourth image with said desktop in response to said selection, receiving an interaction from said user with said other application, and opening said other application in response to said interaction.
Claim 16	
[16]	The system of claim 10, wherein each one of said plurality of applications comprises a web browser application, and each one of said first, second, and third objects comprises a web browser window.
Claim 17	
[17a]	The system of claim 16, wherein said executable code is further configured to: display at least fourth and fifth images in a second 3D space, wherein said fourth image is an image of said plurality of images within said 3D space and said fifth image is an image of at least a portion of a fourth object generated by a fourth application previously opened by said user; and

[17b]	allow said user to interact with said one of said first, second, and third applications by selecting said fourth image, selecting said one of said plurality of images corresponding to said one of said plurality of applications, and interacting with said one of said first, second, and third objects corresponding to said one of said plurality of applications.
Claim 18	
[18]	The system of claim 10, wherein said step of replacing said plurality of images with one of said first, second, and third objects, further comprises enlarging a size of at least said one of said plurality of images before replacing said one of said plurality of images with said one of said first, second, and third objects, thereby simulating movement of said one of said plurality of images in a z-axis of said display device.

Claim 19	
[19pre]	A method for using a two-dimensional (2D) space to selectively interact with at least one of a plurality of applications open on a device and a three-dimensional (3D) space to switch between said plurality of application, comprising:
[19a]	receiving a plurality of inputs from a user, said plurality of inputs comprising at least first, second, and third inputs;
[19b]	opening said plurality of applications in response to said plurality of inputs, said plurality of applications comprising at least first, second, and third applications, wherein for each one of said plurality of applications (i) an output is generated having application-specific data, (ii) displaying said output in said 2D space on said fixed resolution display, and (iii) allowing said user to modify at least a portion of said application-specific data by interacting with said output;
[19c]	allowing a user to view information on said plurality of application, comprising;

[19c-1]	generating at least a plurality of images, said plurality of images comprising at least first, second, and third images, wherein said first image is an image of at least a portion of a first object generated by said first application and having first application-specific data, said second image is an image of at least a portion of a second object generated by said second application and having second application-specific data, and said third image is an image of at least a portion of a third object generated by said third application and having third application-specific data;

[19c-2]	replacing said output corresponding to one of said plurality of applications that is being interacted with by said user in said 2D space with said plurality of images, said plurality of images being displayed in said 3D space and in an order based on a last time that said user one of (i) opened said first application and interacted with said first object, (ii) opened said second application and interacted with said second object, and (iii) opened said third application and interacted with said third object, wherein said 3D space comprises at least a foreground and a background, a first one in said order is displayed in said foreground, and second and third ones in said order are displayed in one of a foreground and a background;
[19c-3]	allowing said user to move said plurality of images, wherein (i) movement of one of said plurality of images results in movement of all of said plurality of images, and
[19c-4]	(ii) continued movement in one direction of one of said plurality of images results in a perception to said user that said one of said plurality of images is moved off of said fixed resolution display; and

[19c-5]	allowing said user to delete at least one of said plurality of images from said 3D space, wherein deletion of said second one in said order results in said third one in said order being moved to a location in said 3D space where said second one in said order was located prior to said deletion; and
[19d]	allowing said user to interact with one of said first, second, and third applications, comprising:
[19d-1]	receiving a selection from said user of one of said plurality of images corresponding to one of said plurality of applications;
[19d-2]	replacing said plurality of images within said 3D space with one of said first, second, and third outputs corresponding to said one of said plurality of applications within said 2D space in response to said selection;
[19d-3]	receiving at least one interaction by said user with said one of said first, second, and third outputs within said 2D space; and
[19d-4]	modifying said one of said first, second, and third application-specific data in response to said at least one interaction.

Claim 20	
[20]	The method of claim 19, wherein said step of replacing said plurality of images with one of said first, second, and third objects, further comprises enlarging a size of at least said one of said plurality of images before replacing said one of said plurality of images with said one of said first, second, and third objects, thereby simulating movement of said one of said plurality of images in a z-axis of said display device.

I. INTRODUCTION

LG Electronics Inc., LG Electronics U.S.A., Inc., and Google LLC (together, "Petitioner") request an Inter Partes Review ("IPR") of claims 1–20 (the "Challenged Claims") of U.S. Patent No. 9,696,868 (the "'868 Patent"). This petition is substantively the same as IPR2023-00344 (which is currently pending institution) and is being filed concurrently with a motion for joinder with respect to that proceeding. Petitioners respectfully request institution of an IPR and cancellation of the Challenged Claims as unpatentable.

The '868 Patent is directed to a three-dimensional computing interface "that allows the user to efficiently navigate though [sic] a virtual space wherein groups of windows can be easily organized, stored, and retrieved." APPLE1001, 1:30-39, 2:41-50. More specifically, the '868 Patent describes an interface that "runs within web browsers (e.g., Internet Explorer and Mozilla Firefox) or as a stand-alone application," and that "simulates a 3-D space within a 2-D window by redrawing objects in the space relative to one another as determined by their perceived distance from the viewer." APPLE1001, 8:10-27. A user of the interface is said to be able to "toggle or switch between 2D and 3D for any selectively captured computing output and information (webpages, applications, documents, desktops or anything that can be visualized on a computer)" by "clicking an icon or bottom

[sic] (analogous to minimize in windows operating system)." APPLE1001, 22:7-30.

The '868 Patent's claimed systems and methods are for "using a two-dimensional (2D) space to selectively interact with a plurality of applications open on a device and a three-dimensional (3D) immersive space to switch between said plurality of application [sic]" were not new. APPLE1001, 38:40-44:33. To the contrary, as demonstrated within this Petition with reference to Dr. Henry Fuchs's testimony and additional evidence, the claimed concepts had already been researched, developed, and integrated into working systems long before the '868 patent's earliest effective filing date.

In fact, the '868 Patent was granted without full consideration to the wide body of applicable prior art, and without a single prior art rejection. *See generally* APPLE1002. And, as Dr. Fuchs explains, the claimed systems and methods would have been obvious to a POSITA based on the teachings of several prior art references. *See infra* §VI; APPLE1003.

For example, Anthony (APPLE1007) describes "[a] three-dimensional (3D) view of a data collection" in the form of a timeline that depicts a plurality of items in focal groups according to a logical order. APPLE1007, Abstract, ¶¶[0042], [0043], [0011], FIG. 4. Each item can be "a visual representation of a file, folder, virtual folder, or any other data object," and a user viewing the timeline can freely

navigate among data objects, and can select particular objects to open associated applications. APPLE1007, ¶¶[0042]-[0044], [0047], [0066], [0084]-[0085], FIG. 5.

Similarly, Hanggie (APPLE1006) describes "[a] method and system for rendering a desktop on a computer that "uses advanced textures, lighting, and 3D transformations, yet supports legacy applications." APPLE1006, ¶¶[0012], [0015], [0026]. For example, Hanggie describes the use of a 3D graphics engine (such as Microsoft Direct3D®) for rendering windows in 3D, and explains that 3D user interaction with a given window can be reported to the corresponding application. Hanggie further explains that a user may manually or automatically switch between 2D and 3D drawing modes based on a number of factors including system capabilities and/or power conservation. APPLE1006, ¶[0092].

By revealing the Challenged Claims as obvious combinations of well-known features, Petitioner seeks to correct the material error that led to issuance of the '868 Patent—the examiner's apparent failure to substantively consider any of the prior art applied in this Petition. Moreover, Petitioner's diligence affords the Board an opportunity to decide patentability at the PTAB before the district court reaches that issue in the co-pending litigation. Indeed, as demonstrated below, this Petition provides compelling reasons for the Board to institute IPR, and to ultimately find the Challenged Claims unpatentable.

II. REQUIREMENTS FOR IPR

A. Standing

Petitioner certifies that the '868 Patent is available for IPR. Petitioner is not barred or estopped from requesting this review. The LG petitioners were served with a complaint regarding the '868 patent less than one year ago. LGE1027.

B. Challenge and Relief Requested

Petitioner requests IPR of the Challenged Claims on the 35 U.S.C. §103 grounds identified in the table below. Accompanying explanations and support are provided in Dr. Fuch's Declaration. APPLE1003, ¶¶[1]-[179].

Ground	Claim(s)	§103 Combination
1A	1-5, 9-14, and 18-20	Anthony in view of Hanggie
1B	6-8, 15-17	Anthony in view of Hanggie and Matthews
1C	7	Anthony in view of Hanggie and Matthews and Robertson
2A	1-5, 9-14, and 18-20	Hanggie in view of Anthony
2B	6-8, 15-17	Hanggie in view of Anthony and Matthews
2C	7	Hanggie in view of Anthony and Matthews and Robertson

The '868 Patent was filed on February 5, 2015. The '868 Patent is a continuation of a number of applications—the first (US Pat. No. 7,735,018) of which was filed on September 13, 2006, which further claims priority from US provisional application No. 60/717,019, filed on September 13, 2005. Each of the references

asserted in this Petition qualifies as prior art under 35 USC §§102(a), (b), or (e) (as shown in the table below) even if the September 13, 2005 date of the provisional application is used as the priority date of the '868 Patent, which Petitioner does not concede.

Reference	Filing Date	Publication Date
US Patent Pub. No. 2005/0088447 ("Hanggie" or APPLE1006)	Oct. 23, 2003	Apr. 28, 2005
US Patent Pub. No. 2005/0091596 ("Anthony" or APPLE1007)	Oct. 23, 2003	Apr. 28, 2005
US Patent Pub. No. 2006/0107229 ("Matthews" or APPLE1008)	Nov. 15, 2004	May 18, 2006
US Patent No. 6,414,677 ("Robertson" or APPLE1009)	Sep. 14, 1998	Jul. 2, 2002

III. OVERVIEW OF THE '868 PATENT

A. Brief Description

The '868 Patent is directed to a three-dimensional (3D) interactive computing interface "that allows the user to efficiently navigate though [sic] a virtual space wherein groups of windows can be easily organized, stored, and retrieved." APPLE1001, 1:30-39, 2:41-50. The '868 Patent's graphical user interface (GUI) "uses the two-dimensional (2D) display of a user's computer to

display three-dimensional (3D) objects in a simulated real-time 3D immersive Cartesian space." APPLE1001, 2:57-62. "Because the 3D GUI creates the illusion of infinite space in 3D, it can create a visual history of the user's computing session, whereby the user can visit past visual computing events (or a snapshot in time) by simply navigating to previously recorded states or viewpoints." APPLE1001, 5:17-32; APPLE1003, ¶[39].

In particular, the 3D GUI program "simulates a 3-D space within a 2-D window by redrawing objects in the space relative to one another as determined by their perceived distance from the viewer. Objects that are supposed to be further away are smaller whereas objects that are supposed to be closer are larger." APPLE1001, 8:25-27. "The program runs within web browsers (e.g., Internet Explorer and Mozilla Firefox) or as a stand-alone application compatible with the local operating system." APPLE1001, 8:10-26; APPLE1003, ¶[40].

Examples of the '868 Patent's GUI are shown in the following figures. FIG. 9 (reproduced below) depicts "a bird's eye view of four 3D stacks 302, 304, 306, 308 drawn in a virtual space 300." APPLE1001, 18:4-27. 3D icons (342, 344, 346, 348) represent four different searches that resulted in the creation of stacks 302, 304, 306, 308. APPLE1001, 29:38-50. Navigator 320 allows a user to navigate through the items (e.g., pages) in each stack. APPLE1001, 29:51-30:7; APPLE1003, ¶[41].

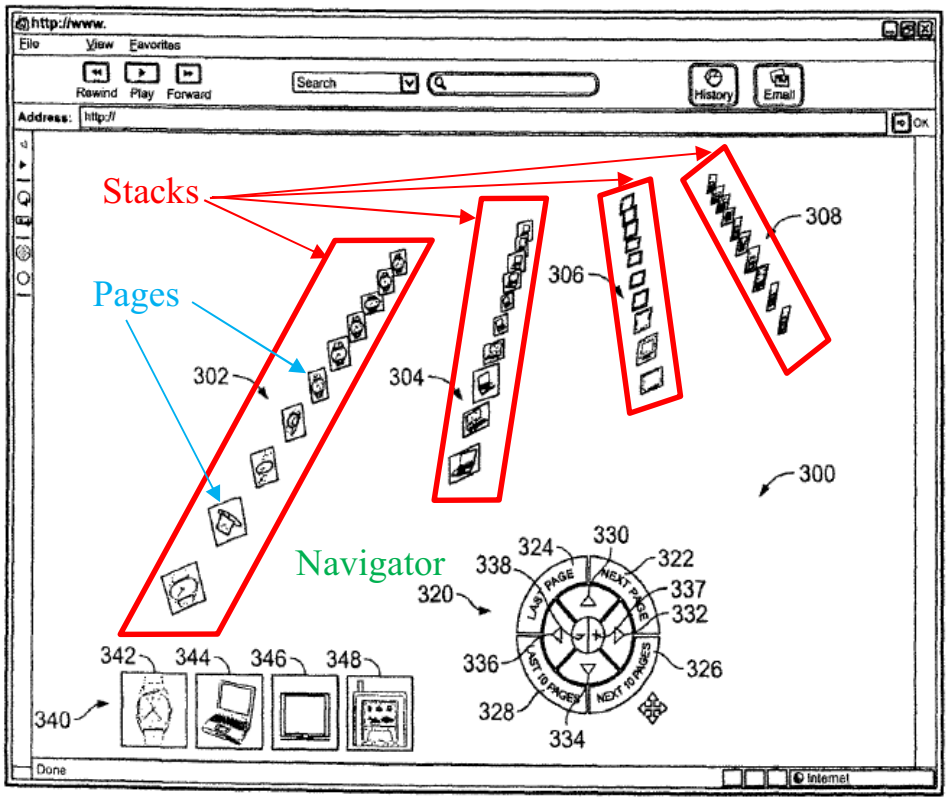


FIG. 9

APPLE1001, FIG. 9[1].

In some cases, as shown below in FIG. 16A, a displayed timeline "includes an icon for each object presented within the virtual space, wherein the icons are organized in linear chronological order according to when the objects were presented within the virtual space." APPLE1001, 3:11-24; APPLE1003, ¶[42].

[1] Annotations to the figures throughout this Petition are shown in color.

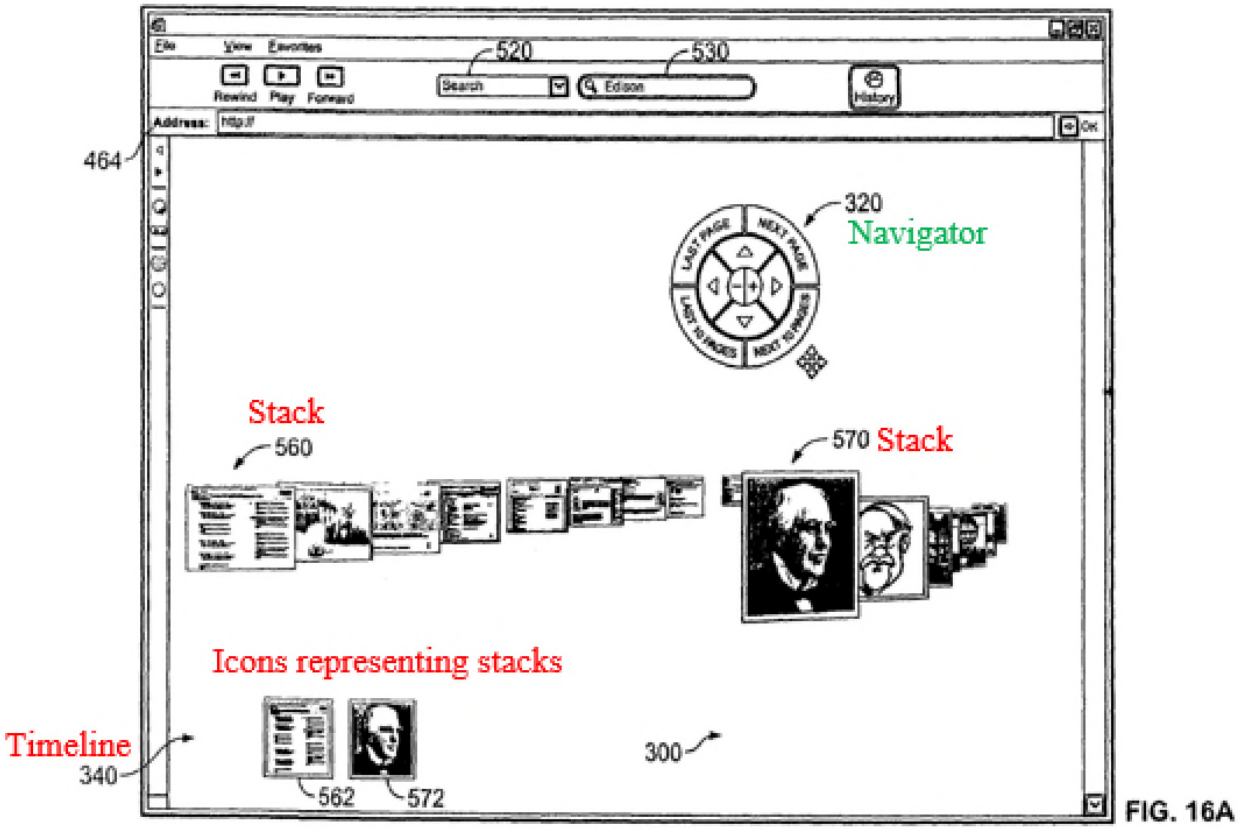


FIG. 16A

APPLE1001, FIG. 16A.

A user of the 3D GUI "can toggle or switch between 2D and 3D for any selectively captured computing output and information (webpages, applications, documents, desktops or anything that can be visualized on a computer) that was drawn within a 3D virtual space at will by" "clicking an icon or bottom (analogous to minimize in windows operating system)." APPLE1001, 22:7-30; APPLE1003, ¶[43].

IV. Level of Ordinary Skill in the Art

A person of ordinary skill in the art relating to the subject matter of the '868 Patent as of September 13, 2005 ("POSITA") would have had at least a Bachelor's degree in electrical engineering, computer science, or a similar discipline, and at least two years of experience in the field working with 2D and 3D graphical user interfaces. APPLE1003, ¶¶[31]-[32]. Superior education could compensate for a deficiency in work experience, and vice-versa. *Id*.

V. CLAIM CONSTRUCTION—37 C.F.R. §42.104(b)(3)

In an effort to promote transparency and consistency between co-pending proceedings, Petitioner notes that the LG petitioners and Patent Owner agreed upon the following construction for the following two terms recited in the '868 Patent:

- 3D space (claims 1, 10, 29): a virtual space defined by a three-dimensional coordinate system; and
- 2D space (claims 1, 10, 29): a finite graphical area defined by a two-dimensional coordinate system.

APPLE1005, 6. The LGE petitioners and PO contest whether the preamble of the independent claims of the '868 Patent are limiting. In this Petition,
the prior art grounds address the preambles, to the extent they are found to be limiting. No other '868 Patent terms are at issue in co-pending district court

litigation. APPLE1003, ¶¶[44]-[45].

VI. THE CHALLENGED CLAIMS ARE UNPATENTABLE

A. GROUNDS 1A/2A: Claims 1-5, 9-14, and 18-20 are obvious over Hanggie in view of Anthony or Anthony in view of Hanggie

1. Anthony[2]

Anthony discloses "[a] three-dimensional (3D) view of a data collection based on an attribute." APPLE1007, Abstract. As shown below in Anthony's FIG. 4 and in view of Anthony's disclosure, the 3D view can take the form of a timeline, which when displayed, depicts a plurality of items in focal groups according to a logical order. APPLE1007, Abstract, ¶¶[0042], [0043], [0011]. The items can be "a visual representation of a file, folder, virtual folder, or any other data object that may be stored in an operating system and/or file system, for example, icons, thumbnails, and the like." APPLE1007, ¶[0042]. The order of the items "may be based on time, thus arranging the items chronologically, e.g., by using a date of creation or date of edit attribute as the ordering attribute." APPLE1007, ¶[0042]; APPLE1003, ¶[46].

[2] General descriptions provided for this and other references and combinations thereof are incorporated into each subsection and mapping of the claims that includes citations to these references.

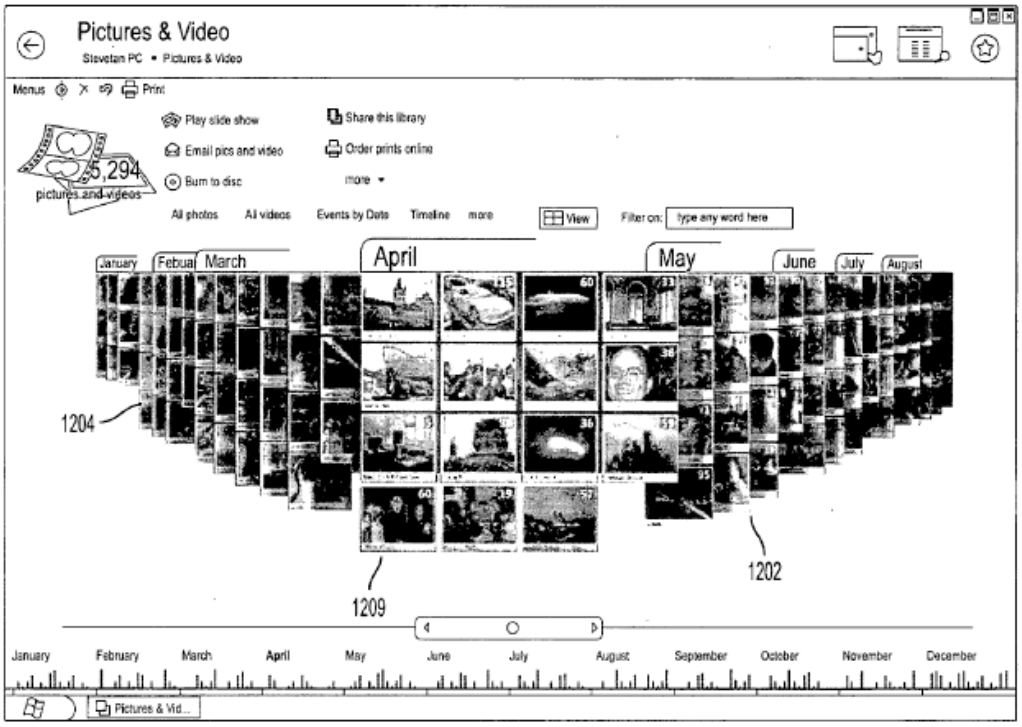


FIG. 4

APPLE1007, FIG. 4.

Anthony's method is implemented in a computing device that includes computer 110 connected to a monitor 191. APPLE1007, FIG. 1, ¶¶[0034]-[0039]. Monitor 191 has a screen with a fixed resolution, and is used to display the timeline. APPLE1007, Abstract, ¶[0050]. An example of the resolution includes a 96 dpi screen resolution. APPLE1007, ¶[0050]. As shown below in FIG. 1, Anthony's computer system includes "application programs 135[/145], other program modules 136[/146], and program data 137[/147]." APPLE1007, ¶¶[0036], [0038]. Anthony's computer system can display images using various application programs, such as Digital Image Pro, preinstalled or installed thereafter

on the operating system. APPLE1007, ¶[0003]. Anthony also gives an example of Excel spreadsheets, indicating that Anthony's computer system can also execute the Microsoft Excel application. APPLE1007, ¶[0049]; APPLE1003, ¶[47].

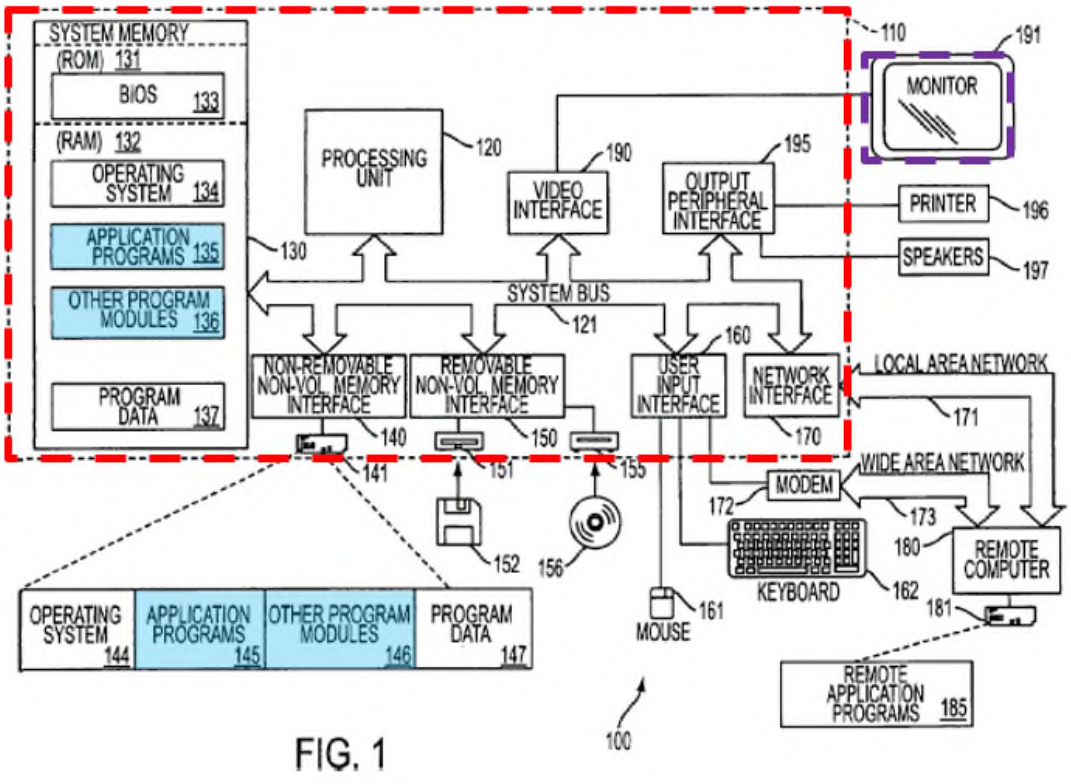


FIG. 1

APPLE1007, FIG. 1.

Anthony's dynamic timeline can be viewed in a 3D environment, as shown above in FIG. 4 in view of the disclosure. APPLE1007, ¶[0043]. A user viewing the timeline can "freely navigate documents, files, or other data objects in a chronological manner, and" can "change the point of focus to an arbitrary location on a timeline." APPLE1007, ¶[0044]; APPLE1003, ¶[48]. For example, as shown in FIG. 5:

> [a] timeline 500 has a start date 502 and an end date 504. The timeline 500 may include various time intervals (e.g., days, months, years, etc., not called out in FIG. 5). … the point of view of the GUI may be based on a conceptual camera 506 pointed at the time interval containing the files or objects on which the user wishes to focus. This time range is called the focal point 508 or focal point interval. The focal point 508 may represent a point in time or an interval of time on the timeline 500. Items that are within the time range covered by the focal point are collectively called the focal group 509. Items in the focal group 509 are typically presented to the user in greater detail than is shown for items in other time intervals on the timeline 500. Items in other time intervals on timeline 500 are grouped together in non-focal groups 510.

APPLE1007, ¶[0044].

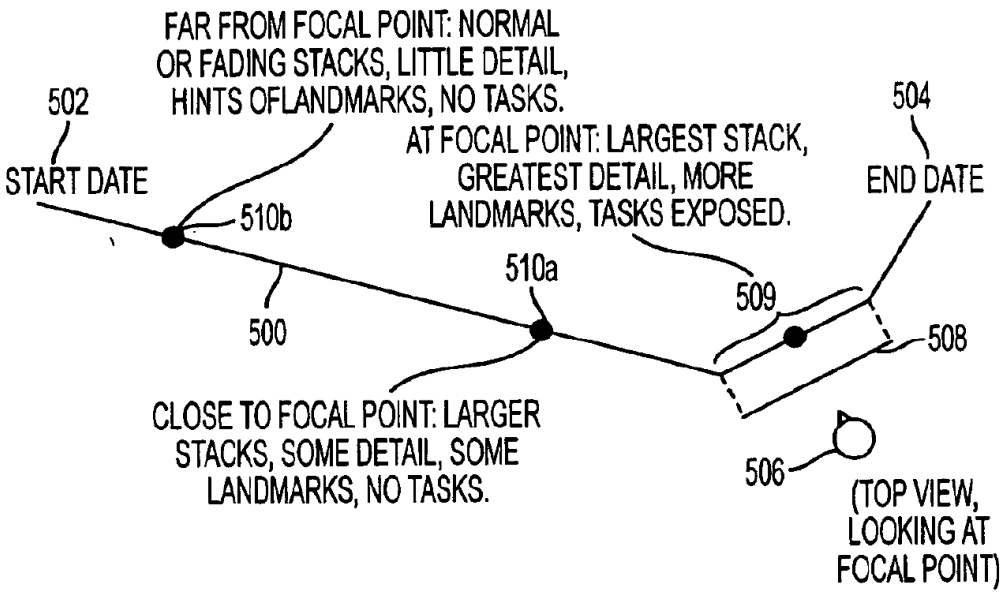


FIG. 5

APPLE1007, FIG. 5.

A user can interact with the dynamic timeline (shown in a 3D environment) by navigating to different points of the timeline or selecting a particular icon to open the application associated with the icon. APPLE1007, ¶¶[0043]-[0044], [0047], [0066], [0084]-[0085]; APPLE1003, ¶[49].

2. Hanggie

Hanggie discloses "rendering a desktop on a computer using a composited desktop model operating system" "with intrinsic support for composition-agnostic legacy applications." APPLE1006, Abstract, ¶[0001]. Hanggie discloses a *compositing* Desktop Window Manager (CDWM) that "provide[s] a rich, full featured operating system … to provide a desktop that uses advanced textures, lighting, and 3D transformations, yet supports legacy applications." APPLE1006,

¶¶[0012], [0015], [0026], [0007], [0005]; APPLE1003, ¶[50].

Hanggie's CWDM can be implemented using computer software and systems, as shown in Hanggie's FIGS. 1A and 1B. For instance, Hanggie's FIG. 1A (reproduced below) illustrates an example computer system environment 100 including computer 110 that further includes a processing unit 120, a system memory 130, and a system bus 121. APPLE1006, ¶¶[0028]-[0031]. The system memory 130 includes read only memory (ROM) 131 and random access memory (RAM) 132 that stores operating system 134, application programs 135, other program modules 136, and program data 137. APPLE1006, ¶¶[0033]-[0034]. A hard disk drive can store the same or different data than the RAM 132. APPLE1006, ¶[0035]; APPLE1003, ¶[51].

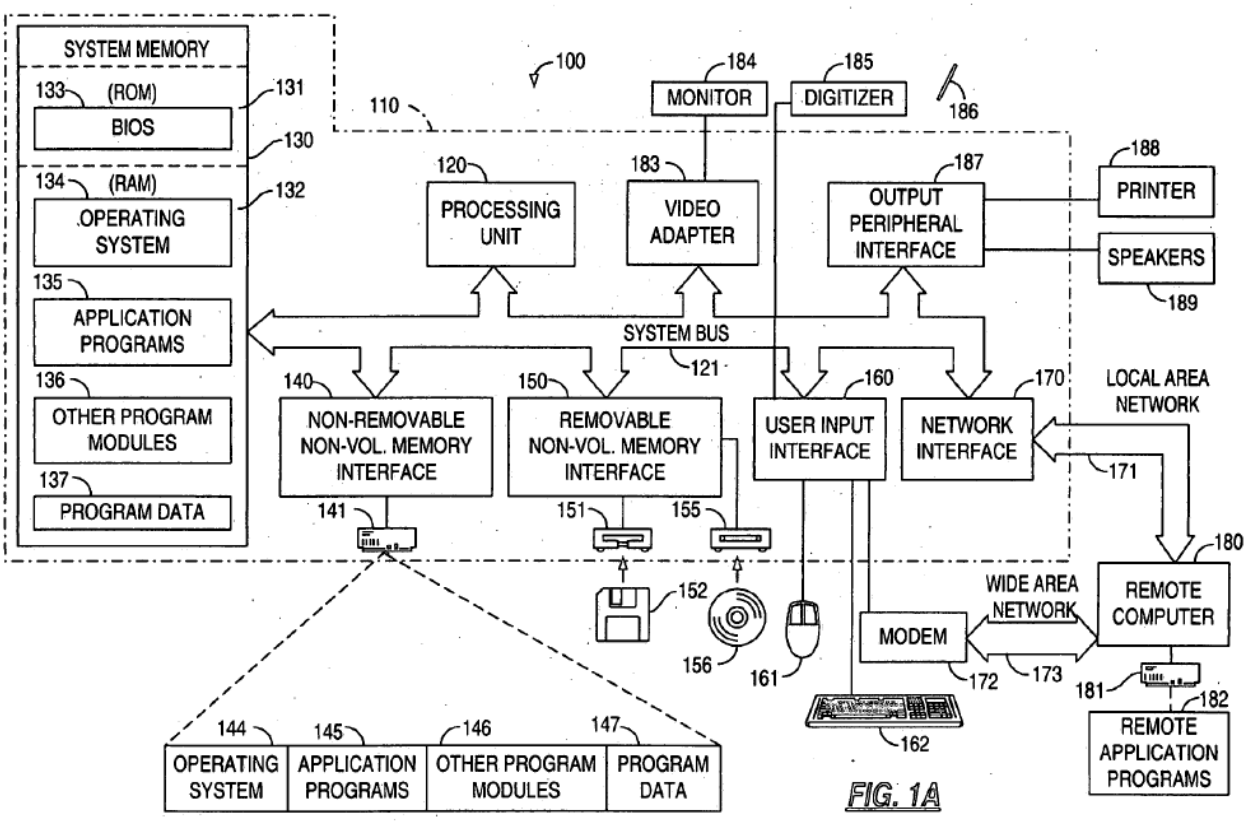


FIG. 1A

APPLE-1006, FIG. 1A.

"The CDWM composes the desktop … beginning with the desktop background and proceeding through overlapping windows in reverse Z order. While composing the desktop, the CDWM may draw each window based in part on the content on top of which the window is being drawn, and based in part on other environmental factors (e.g., light source, reflective properties, etc.)." APPLE1006, ¶[0039]. "The CDWM may reside as part of the operating system 134, 144," and may use a compositing engine based on Direct3D® and DirectX® technology by Microsoft Corporation or OpenGL graphics engine. APPLE1006, ¶[0040]. The compositing engine "enables 3D graphics and animation,

transparency, shadows, lighting effects, bump mapping, environment mapping, and other rich visual features on the desktop." APPLE1006, ¶[0040]; APPLE1003, ¶[52].

Referring to Hanggie's FIG. 1B (reproduced below), the CWDM 190 "may include an Application Programming Interface 190a through which a composition-ware Application software 191 obtains CDWM window and content creation and management services" and is connected to a Legacy Windowing Graphics Subsystem 192. APPLE1006, ¶[0041]. "The Legacy Graphical User Interface Subsystem 192 may include a Legacy Window Manager 192a and Legacy Graphics Device Interface 192b." APPLE1006, ¶[0042]. "The Legacy Graphics Device Interface 192b provides 2D graphics services to both legacy applications as well as the Legacy Window Manager." APPLE1006, ¶[0042]; APPLE1003, ¶[53].

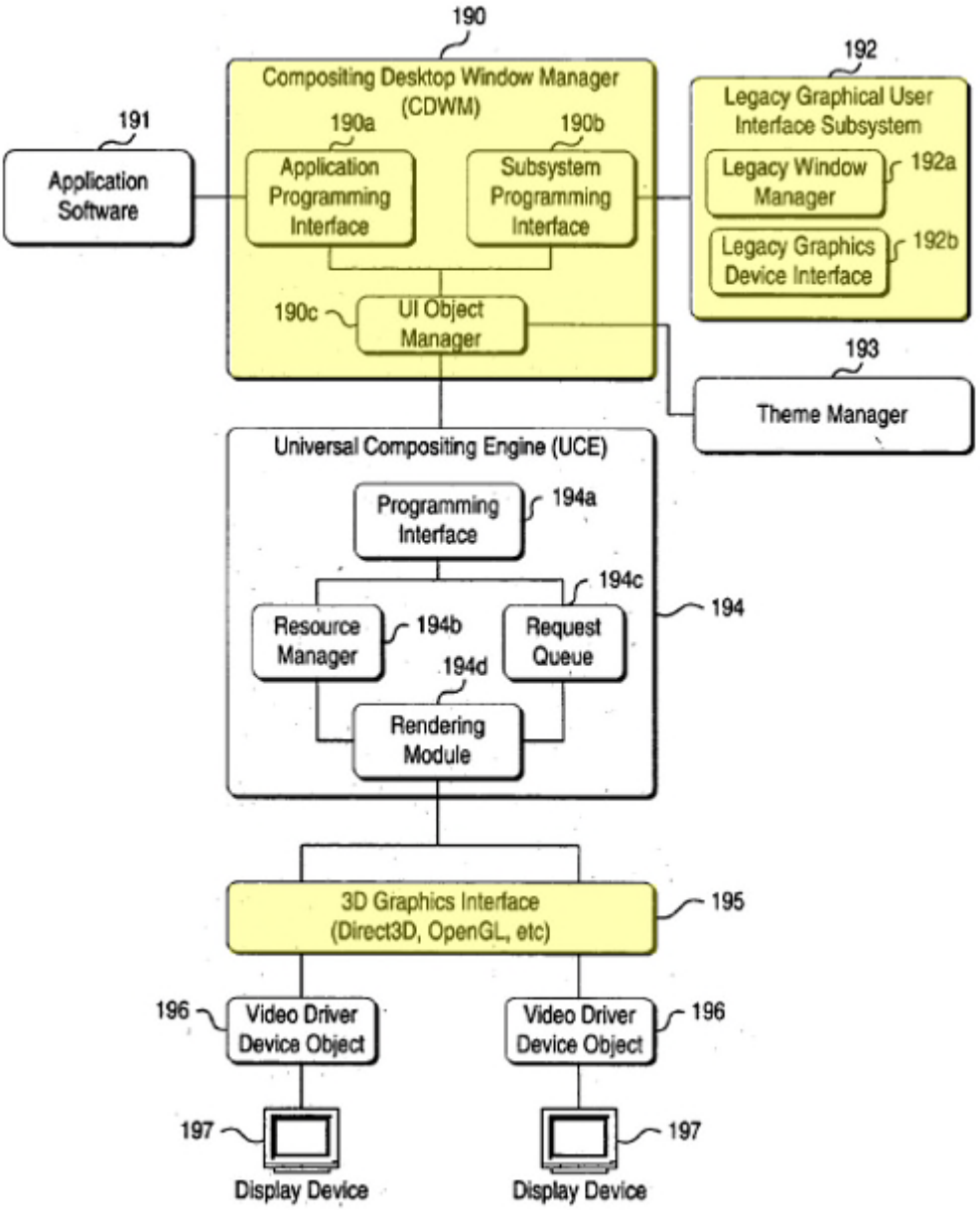


FIG. 1B

APPLE1006, FIG. 1B.

As shown in FIG. 1B, a Unified Compositing Engine (UCE) 194 is connected to the CDWM 190 and a 3D Graphics Interface 195, and provides graphic services to the CDWM 190. APPLE1006, ¶¶[0043]-[0044]. The 3D Graphics Interface 195 can include software services "such as Direct3D®,

OpenGL®, or the like" and provide an abstract interface to display devices 197 in a multiple-display system via multiple device drivers 196. Thus, Hanggie's system can support 2D displays of legacy applications as well as 3D displays. APPLE1003, ¶[54].

With respect to the rendering of windows in 3D, Hanggie explains that "the UCE processes incoming composition and resource updates from the CDWM," and uses "the immediate-mode rendering services provided by a **3D graphics engine** (such as Microsoft Direct3D®), which in turn transfers the desktop to a primary display surface."[3] APPLE1006, ¶¶[0056], [0065]. An example of such a window is shown below in Hanggie's FIG. 3. Hanggie explains that "[t]he frame or base content 303 of the window 301 may host child content including **buttons 305 (e.g., used to restore, maximize, minimize, close the window, etc.),** an indicative icon 307, scrollbars 309, menu bar 311, and window caption text 313. A primary content object area 315 may be derived from the redirection buffer obtained from the Legacy Window and Graphical User Interface subsystem, or be created and attached to the standard base content and rendered by a composition-aware owning application." APPLE1006, ¶[0058]; APPLE1003, ¶[55].

[3] Throughout this Petition, bolding in quotations is added for emphasis, unless otherwise indicated.

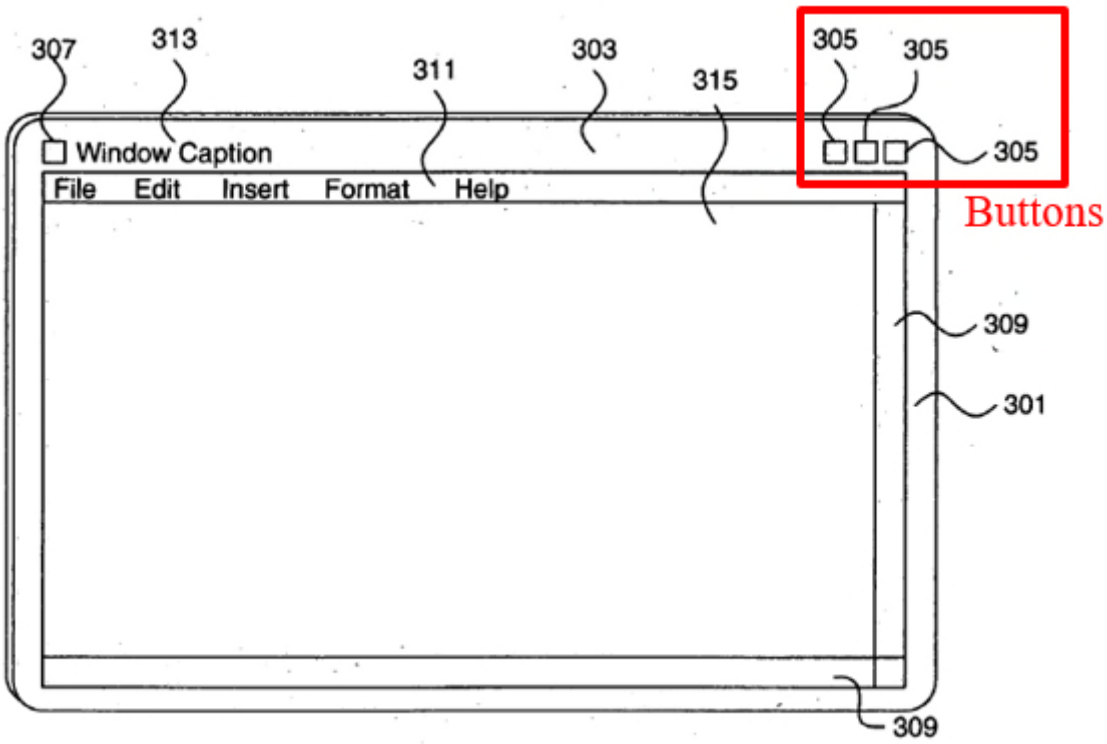


FIG. 3

APPLE1006, FIG. 3.

Another example of a window is shown in FIG. 5, which depicts a "My Documents" folder/window within which icons for additional folders/applications (e.g., My Music, My Pictures) are located. APPLE1006, FIG. 5, ¶[0083]; APPLE1003, ¶[56].

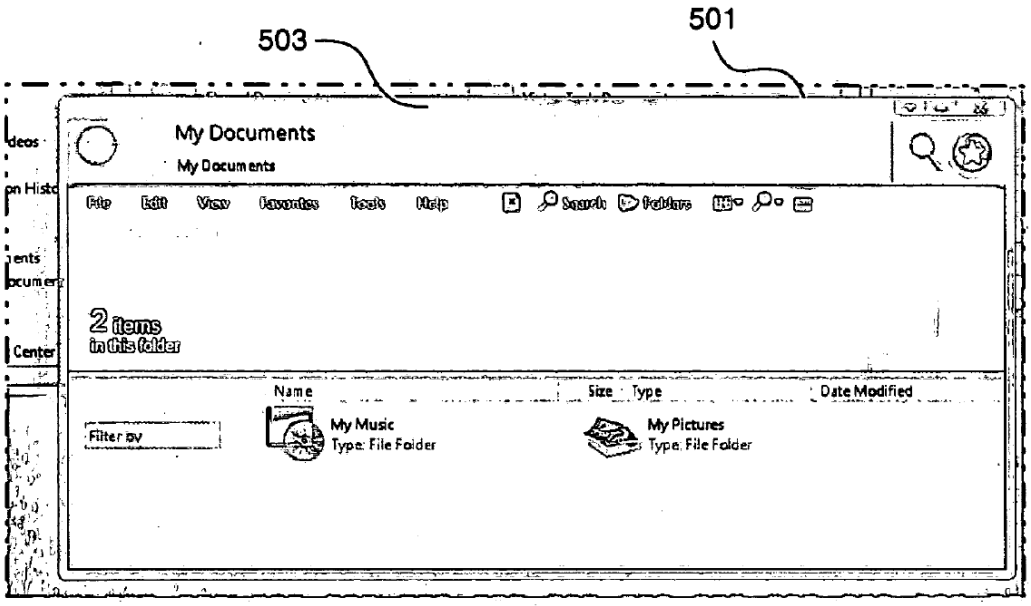


FIG. 5

APPLE1006, FIG. 5.

The window may be provided by or correspond to an application. APPLE1006, ¶[0058]. "[T]he CDWM may allow an application to alter the default anatomy of a window by allowing applications to define multiple custom content objects." APPLE1006, ¶[0059]. Hanggie teaches that each CDWM window can be defined by attributes related to an application, such that 3D user interaction with the window can be reported to the application. APPLE1006, ¶[0060]; APPLE1003, ¶[57].

Hanggie explains that a user may choose to operate in a 2D mode (legacy drawing mode) or a 3D mode (CDWM) based on a number of factors such as system capabilities or power conservation conditions. APPLE1006, ¶[0092]. For example, a user may want to the computer to operate in the legacy drawing mode

when the computer has "low video memory" or "no 3D acceleration hardware." APPLE1006, ¶[0092]. Hanggie explains that "the selection may be made automatically or manually. For example, the selection may be made via manual user control, in accordance with the drawing mode defined by an activated visual style (theme) selected by the user." APPLE1006, ¶[0092]; APPLE1003, ¶[58].

3. Combination of Anthony and Hanggie

As explained in more detail below, a POSITA would have been motivated and found it obvious to combine the teachings of Anthony and Hanggie to: (1) implement one or more windows corresponding to one or more applications as displayed focal group windows in Anthony's timeline; (2) implement control buttons 305 on application windows within the timeline that would allow a user to restore, maximize, minimize, or close the windows; and (3) implement a control selection through which a user can switch between 2D and 3D viewing of the application windows. APPLE1003, ¶[59].

In more detail, and as shown in Anthony's FIG. 4 (reproduced below), Anthony's timeline depicts items in focal groups according to a logical order, and the depicted items visually represent "a file, folder, virtual folder, or **any other data object** that may be stored in an operating system and/or file system, for example, icons, thumbnails, and the like" (emphasis added). APPLE1007, ¶[0042]; *see also id.*, Abstract, ¶¶[0043], [0011]; APPLE1003, ¶[60].

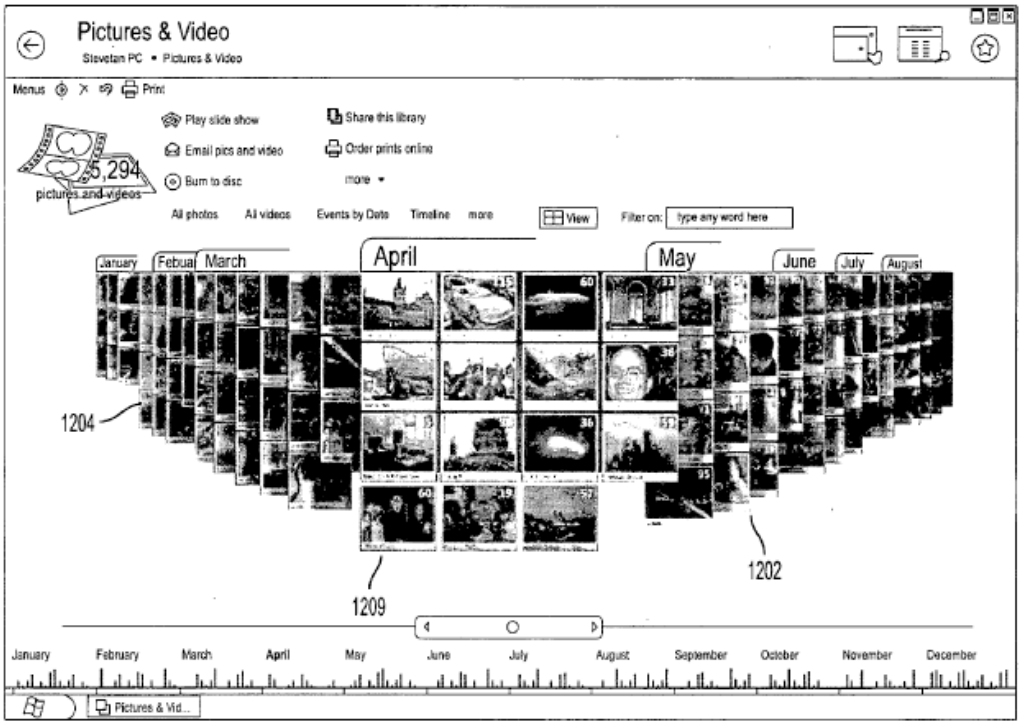
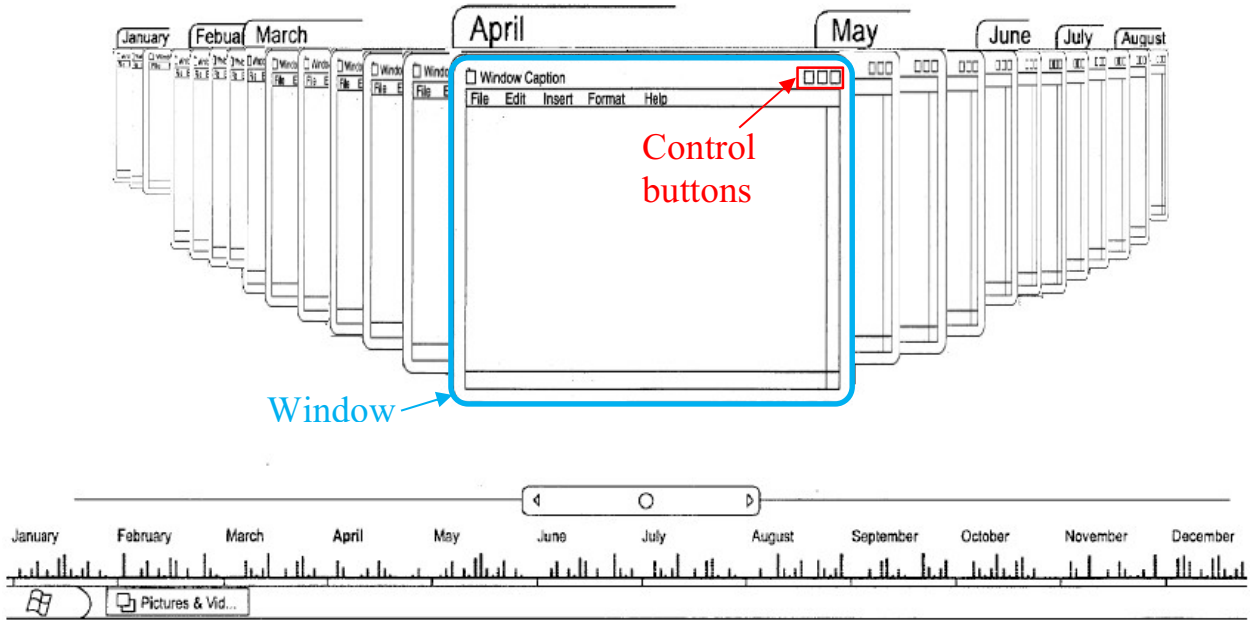


FIG. 4

APPLE1007, FIG. 4.

Consistent with Anthony's description of visually representing "any…data object" in its dynamic timeline, a POSITA would have understood and found it obvious that a plurality of application windows could be organized and represented within Anthony's timeline. APPLE1003, ¶[61]; APPLE1007, ¶[0042]. As shown in the exemplary figure reproduced below, a POSITA would have been motivated to combine Anthony and Hanggie's teachings such that the focal group windows depicted in Anthony's FIG. 4 are replaced by application windows that are similar to those shown in Hanggie's FIG. 3, and that incorporate several of the features described by Hanggie. APPLE1003, ¶[61].



APPLE1003, pg. 39.

For example, consistent with Anthony's teachings, the POSITA would have found it obvious to organize the plurality of application windows within a dynamic timeline and, consistent with Hanggie's teachings, each of the plurality of application windows would feature a window caption and control buttons, with the buttons enabling selection between 2D and 3D viewing modes. APPLE1003, ¶¶[61]-[62]; APPLE1007, Abstract, ¶¶[0042], [0043], [0011], FIG. 4; APPLE1006, ¶[0058], FIG. 3.

A POSITA would have been motivated to organize and represent application windows in this manner for many reasons. APPLE1003, ¶¶[62]-[66].

For instance, the POSITA would have understood and found it obvious that

window captions indicating the name of the application with which each window is associated would be consistent with user expectations, would be convenient to the user, and would assist the user in navigating between applications represented within the timeline. APPLE1003, ¶[61]; APPLE1006, ¶¶[0056], [0062], FIG. 3. For example, a window caption for a window corresponding to one or more Microsoft Excel spreadsheets could be "Microsoft Excel," while a window caption for a window corresponding to one or more Microsoft Word documents could instead be "Microsoft Word." *See* APPLE1007, ¶[0049]; APPLE1006, FIG. 3; APPLE1003, ¶[63].

A POSITA would also have recognized that implementing Hanggie's control buttons 305 in Anthony's focal group windows such that the user could close, minimize, maximize, or restore a window when desired by selecting a corresponding button would be consistent with user expectations, would be convenient to the user, and would assist the user in interacting with applications represented within the timeline. APPLE1003, ¶[60]; APPLE1006, ¶[0056], FIG. 3. Control buttons of this nature would have been understood to provide a user with more control over the information displayed within the timeline and application windows, and to allow the user to view or interact with a particular window only when desired. APPLE1003, ¶[60].

Consistent with these advantages, with Hanggie's teachings, and with

expectations surrounding the Microsoft Windows environments described by each of Anthony and Hanggie, a POSITA would have understood and found it obvious that a maximized window would occupy all, or almost all, parts of a display. APPLE1003, ¶¶[61], [106]. As such, a POSITA would have understood that viewing a maximized window in a traditional 2D format would, in some cases, allow certain users to interact with an application in a 2D environment similar to their past experiences in 2D. APPLE1003, ¶[61]; APPLE1006, ¶[0092].

For at least that reason, the POSITA would have found it obvious to implement the maximize control button such that manual user selection of that button would result in Anthony's user interface switching from a 3D timeline view enabling user interaction with the plurality of applications represented within windows displayed within that timeline, to a 2D view of the maximized window. APPLE1003, ¶[61]; APPLE1006, ¶[0092].

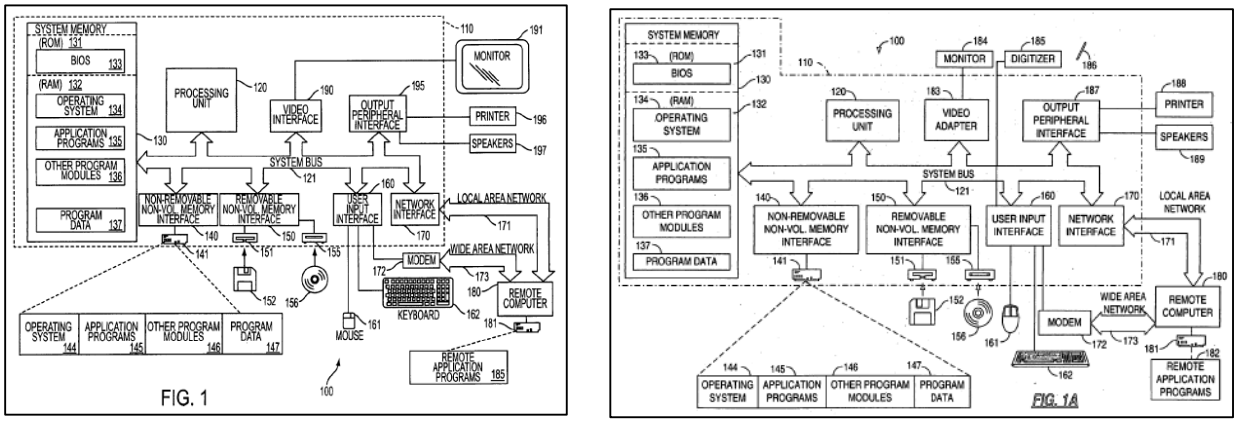
Additional advantages would have further motivated the POSITA to implement the maximize control button in this manner. APPLE1003, ¶[62]. For example, because a maximized 2D window would obscure Anthony's 3D timeline, it would not be necessary to expend computing resources on rendering the 3D timeline while a window is in a maximized state. APPLE1003, ¶[62]. For example, Hanggie teaches that it can be beneficial to switch from a 3D mode to a 2D mode to reduce graphics processing and thereby conserve battery power, and a

POSITA would have recognized that those same advantages could be obtained by switching from a 3D to a 2D state upon user selection of a maximize control button. APPLE1003, ¶[62]; APPLE1006, ¶[0092].

A POSITA would have reasonably expected success in combining Anthony and Hanggie in the above-noted manner, at least because doing so would have involved nothing more than combining prior art elements according to known methods to yield predictable results. APPLE1003, ¶[64]. For example, the combination of Hanggie and Anthony would yield the predictable results of Anthony's system providing standard captions and control buttons for application windows represented within its 3D timeline, and being able to switch between 3D and 2D interface modes. APPLE1003, ¶[64].

Moreover, the combination would have been foreseeable and predictable as Anthony and Hanggie are directed to similar systems and use the same or similar software for rendering 3D graphics. APPLE1003, ¶[64]. For instance, as shown below, Anthony's computer system, as depicted in Anthony's FIG. 1, is highly similar to Hanggie's computer system shown in Hanggie's FIG.1A (e.g., compare Hanggie and Anthony's system memory 130, hard disk drive 141, processing unit 120, monitor 191/184). Further, Anthony and Hanggie both disclose using the same or similar 3D Graphics software such as Direct3D®, or OpenGL® to enable 3D viewing of objects (*see* Anthony's ¶[0043] ("The 3D environment may be

implemented utilizing graphics technology such as DirectX®, Direct3D®, OpenGL®, GDI, a media integration layer, or some other presentation platform as is known in the art. In some embodiments, 3D objects (which contain 3D properties) may be utilized to provide the 3D effect") and Hanggie's ¶[0047] ("The 3D Graphics Interface 195 may include a … graphics service such as Direct3D®, OpenGL®, or the like.)" APPLE1003, ¶[64].

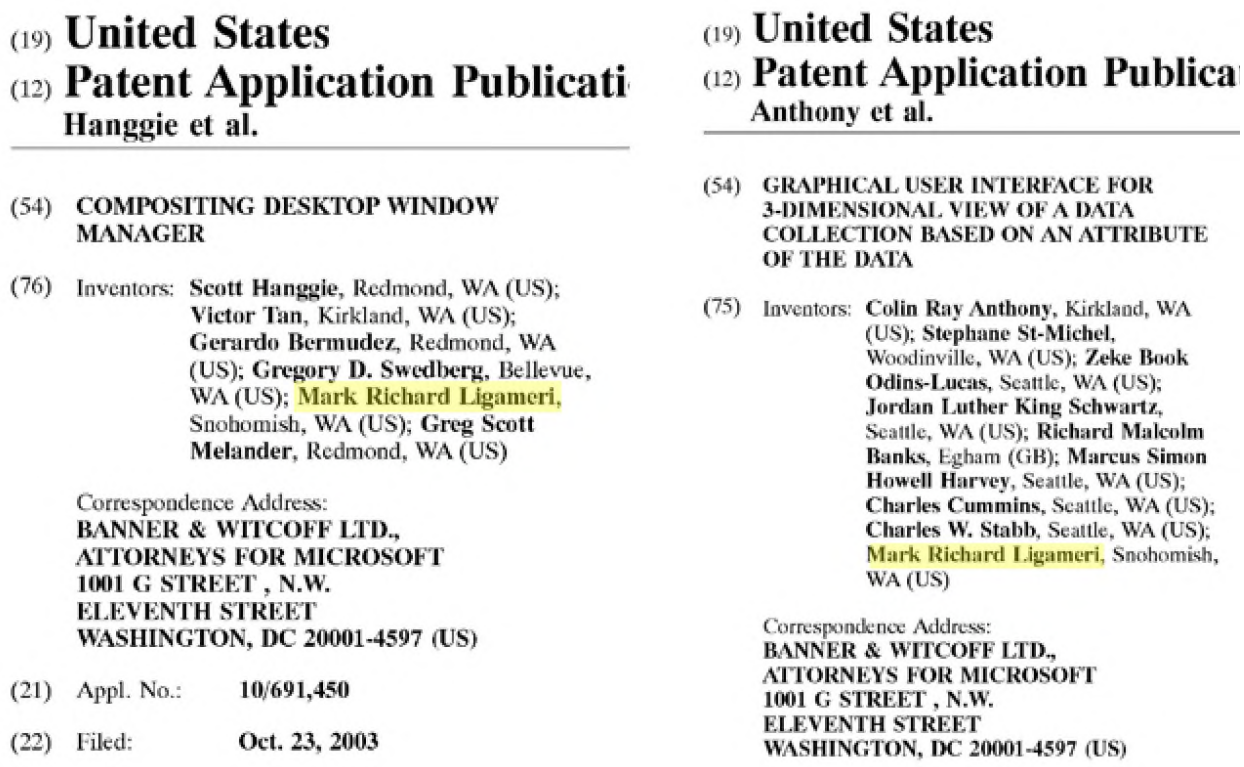


APPLE1007, FIG. 1 (left); APPLE1006, FIG. 1A (right).

As shown in Anthony's FIG. 1 and Hanggie's FIG. 1A, Anthony and Hanggie's systems have similar architectures. Several components, such as the system memory 130, computer 110, and processing unit 120, even share the same reference numbers. And because the functions and operations of Anthony and Hanggie's system components are largely similar (if not identical), a POSITA would have understood that descriptions associated with components shared commonly in both systems would have been obvious to implement in the Anthony and Hanggie combination. APPLE1003, ¶[65]. For example, a POSITA would

have understood that features of components such as input and output devices (e.g., keyboard, monitor, printer) described by either reference could be readily implemented as part of a 3D graphics rendering system combining Anthony and Hanggie's teachings. APPLE1003, ¶[65].

Anthony and Hanggie also share one common inventor and a common assignee, and are both directed to graphical user interfaces and systems for 3D viewing of data. *See below* APPLE1006, cover, title, abstract; APPLE1007, cover title, abstract. These and other commonalities between Anthony and Hanggie would have led a POSITA to have a reasonable expectation of success in combining the teachings of the two references, as described above. APPLE1003, ¶[66].



(19) **United States**
(12) **Patent Application Publicati**
Hanggie et al.

(54) **COMPOSITING DESKTOP WINDOW MANAGER**

(76) Inventors: **Scott Hanggie**, Redmond, WA (US); **Victor Tan**, Kirkland, WA (US); **Gerardo Bermudez**, Redmond, WA (US); **Gregory D. Swedberg**, Bellevue, WA (US); Mark Richard Ligameri, Snohomish, WA (US); **Greg Scott Melander**, Redmond, WA (US)

Correspondence Address:
BANNER & WITCOFF LTD.,
ATTORNEYS FOR MICROSOFT
1001 G STREET , N.W.
ELEVENTH STREET
WASHINGTON, DC 20001-4597 (US)

(21) Appl. No.: **10/691,450**

(22) Filed: **Oct. 23, 2003**

(19) **United States**
(12) **Patent Application Publica**
Anthony et al.

(54) **GRAPHICAL USER INTERFACE FOR 3-DIMENSIONAL VIEW OF A DATA COLLECTION BASED ON AN ATTRIBUTE OF THE DATA**

(75) Inventors: **Colin Ray Anthony**, Kirkland, WA (US); **Stephane St-Michel**, Woodinville, WA (US); **Zeke Book Odins-Lucas**, Seattle, WA (US); **Jordan Luther King Schwartz**, Seattle, WA (US); **Richard Malcolm Banks**, Egham (GB); **Marcus Simon Howell Harvey**, Seattle, WA (US); **Charles Cummins**, Seattle, WA (US); **Charles W. Stabb**, Seattle, WA (US); Mark Richard Ligameri, Snohomish, WA (US)

Correspondence Address:
BANNER & WITCOFF LTD.,
ATTORNEYS FOR MICROSOFT
1001 G STREET , N.W.
ELEVENTH STREET
WASHINGTON, DC 20001-4597 (US)

APPLE1006, cover (left); APPLE1007, cover (right).

4. Combination of Hanggie and Anthony

As explained in more detail below, a POSITA also would have been motivated and found it obvious to combine Hanggie and Anthony such that the teachings of Anthony are implemented in Hanggie's system. APPLE1003, ¶[67].

For example, Hanggie describes rendering application windows similar to the window shown in Hanggie's FIG. 3 (reproduced below), and in doing so explains that its "CDWM may draw each window based in part on the content on top of which the window is being drawn," and that the windows may be arranged in a stacked order. APPLE1006, ¶¶[0039], [0003]. However, Hanggie does not explicitly describe how the stacked order of windows can be can be arranged in a

user-friendly manner to facilitate user interaction. A POSITA would have recognized that Anthony provides those details with respect to the chronological ordering of windows in Anthony's timeline, which may be based, for example, on "date of creation or date of edit attribute as the ordering attribute." APPLE1003, ¶[67]; APPLE1007, ¶[0042].

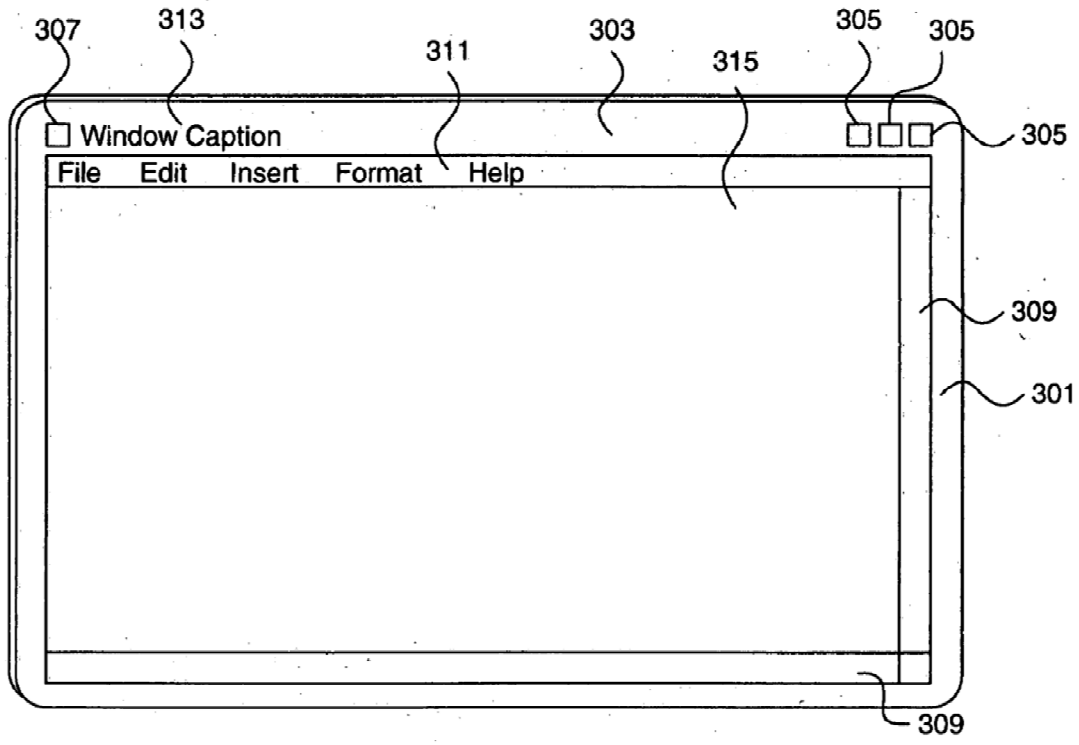
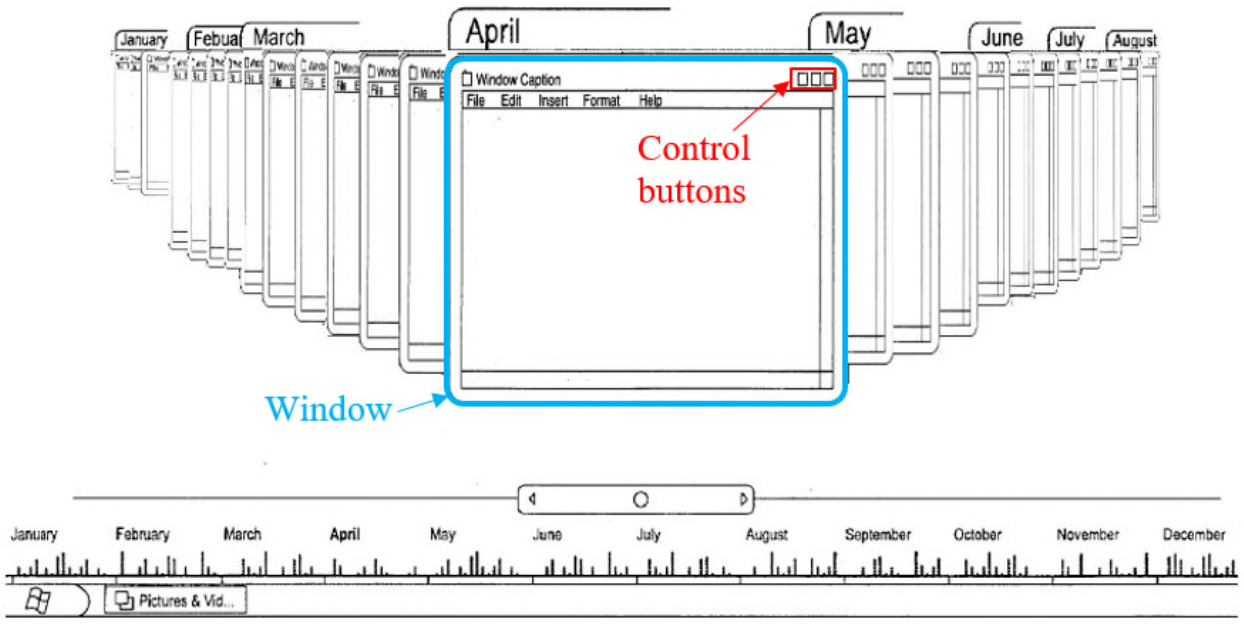


FIG. 3

APPLE1006, FIG. 3.

Consistent with Hanggie and Anthony's descriptions, a POSITA would have been motivated to combine Hanggie and Anthony's teachings such that Hanggie's application windows could be organized and represented within a 3D timeline

similar to Anthony's timeline. APPLE1003, ¶[68]; APPLE1007, ¶[0042], FIG. 4. Indeed, the POSITA would have recognized that ordering and representing Hanggie's windows in this manner would allow for efficient, intuitive, and user-friendly navigation. APPLE1003, ¶[68]. Indeed, the Hanggie-Anthony combination would result in a similar system and interface (shown below) as described above in §VI.A.3.



APPLE1003, pg. 44.

Moreover, the POSITA would have found it obvious to implement Hanggie's maximize control button such that manual user selection of that button would result in Hanggie's user interface switching from a 3D timeline view, to a 2D view of the maximized window. APPLE1003, ¶[69]; APPLE1006, ¶[0092]. Indeed, the same advantages to implementing a maximize button described in

§VI.A.3 in connection with the Anthony-Hanggie combination apply equally to the Hanggie-Anthony combination.

A POSITA would have reasonably expected success in combining Hanggie and Anthony in the above-noted manner at least because doing so would have involved nothing more than combining prior art elements according to known methods to yield predictable results. APPLE1003, ¶[70]. For example, applying Anthony's teachings to Hanggie's system would have yielded the predictable result of Hanggie's windows being arranged chronologically in a timeline, thereby providing a logical, intuitive, and user-friendly interface for 3D navigation between multiple windows in Hanggie's system. APPLE1007, ¶[0042], FIG. 4; APPLE1003, ¶[70].

Moreover, the combination would have been foreseeable and predictable at least because, as explained above in §VI.A.3, Anthony and Hanggie are directed to similar systems, describe similar methods, disclose the use of the same or similar software for rendering 3D graphics, share a common inventor, and share an assignee. APPLE1006, ¶[0047], cover, title, abstract; APPLE1007, ¶[0043], cover, title, abstract; APPLE1003, ¶[71].

As noted above, the combination of Hanggie and Anthony and the combination of Anthony and Hanggie would have resulted in the same features being combined in a similar manner to yield a combined system (hereinafter

referred to as "**HAC**"). APPLE1003, ¶[72]. Thus, although separate combinations are identified in this Petition, the mapping of the prior art references to the Challenged Claims presented below in §VI.A.5 is the same under both combinations. APPLE1003, ¶[72]. In addition, although a single reference may be referenced in the mappings detailed in §§VI.A.5, VI.B.3, VI.C.3, the single reference is part of the HAC system described in this section. APPLE1003, ¶[73].

5. Manner in which Grounds 1A and 2A render obvious Claims 1-10 and 13-30

Claim 1

[1pre]

HAC's computer system 110 implements a method in which a monitor with a fixed resolution displays a 2D space featuring windows that enable a user to selectively interact with a plurality of open applications, and in which the same monitor displays a 3D immersive space enabling the user to switch between the plurality of applications. APPLE1003, ¶[86].

For example, Hanggie's CDWM 190 "provide[s] a rich, full featured operating system" enabling both 2D and 3D display of windows within "a desktop that uses advanced textures, lighting, and 3D transformations," and Anthony describes a 3D immersive space for presenting windows within a logically-ordered dynamic timeline view. APPLE1006, ¶¶[0012], [0015], [0026], [0039], [0040], [0041], FIG. 3; APPLE1007, APPLE1007, Abstract, ¶¶[0042], [0043], [0011],

FIG. 4. As further explained in §§VI.A.3-VI.A.4, a POSITA would have been motivated and found it obvious to combine these teachings in HAC such that the windows described by Hanggie are presented within Anthony's dynamic timeline view. APPLE1003, ¶[86].

In more detail, HAC enables a user to display a plurality of windows, respectively associated with a plurality of applications, in a 2D legacy drawing mode, with 2D graphics provided by the Legacy Graphics Device Interface 192b[4] described by Hanggie with respect to its Legacy Graphical User Interface subsystem. APPLE1006, ¶¶[0092], [0042]; *see supra* §VI.A.2, *infra* [1b]. Hanggie's FIG. 3 (reproduced below) depicts an example of such a window.

[4] Consistent with the construction of "2D space" noted in §V ("a finite graphical area defined by a two-dimensional coordinate system"), Hanggie's Legacy Graphics Interface 192 is explicitly described as "provid[ing] 2D graphics services" to legacy applications and a Legacy Window Manager, and for at least that reason would have been understood by a POSITA to provide a 2D space. APPLE1006, ¶[0042]; APPLE1003, pg. 54.

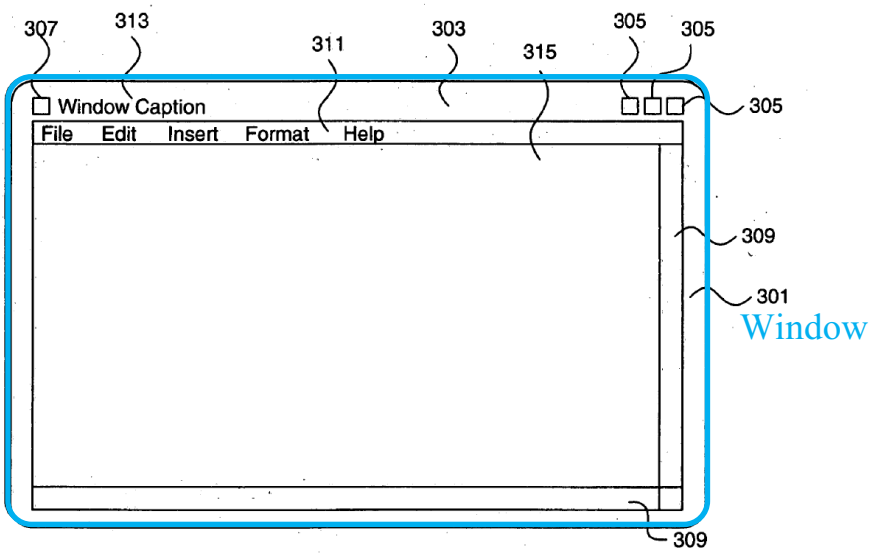


FIG. 3

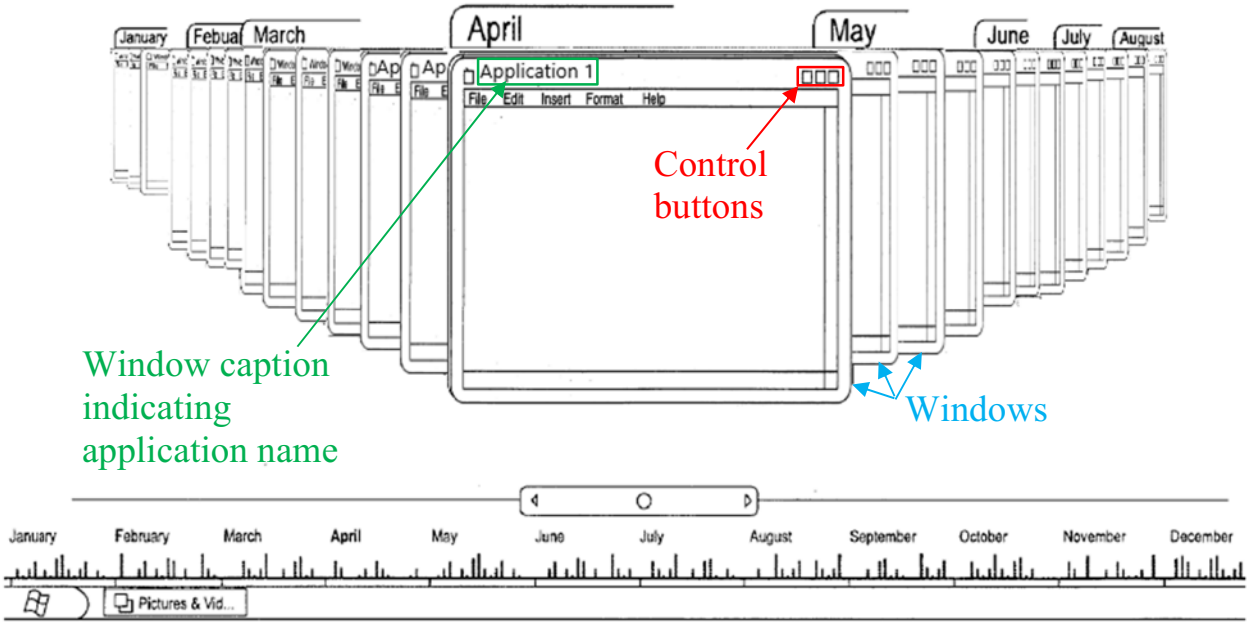
APPLE1006, FIG. 3.

As shown, each window 301 includes buttons 305 (e.g., used to restore, maximize, minimize, and/or close the window), scrollbars 309, menu bar 311, and primary content object area 315 to facilitate user interaction with the window and corresponding application. APPLE1006, ¶[0058], FIG. 3; APPLE1003, ¶[86].

And, as explained in §§VI.A.3-VI.A.4 and shown in the figure below, in HAC, Hanggie's windows can be displayed/rendered in a three-dimensional (3D)[5]

[5] Consistent with the construction of "3D space" noted in §V ("a virtual space defined by a three-dimensional coordinate system"), Anthony's timeline is explicitly described as being implemented in a 3D environment using 3D graphics

immersive space such as Anthony's dynamic timeline view. APPLE1007, ¶[0042]; APPLE1003, ¶¶[87].



APPLE1003, pg. 56.

Further, a POSITA would have understood and found it obvious that HAC's 191/184 monitor[6] has a fixed resolution. APPLE1003, ¶[88]. For instance,

---

technology, and for at least that reason would have been understood by a POSITA to provide a 3D space. APPLE1007, ¶[0043]; APPLE1003, pg. 55.

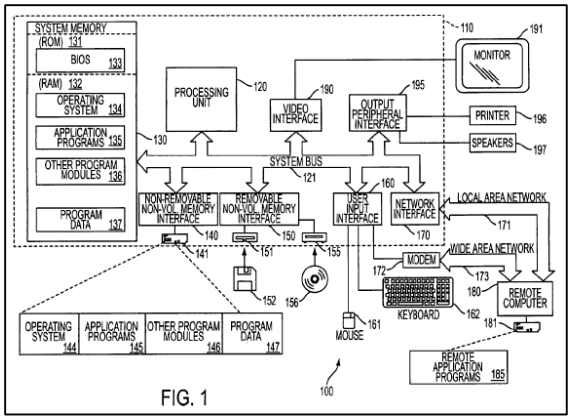
[6] As explained above in §§VI.A.3-VI.A.4, Hanggie and Anthony describe systems having similar architectures, and a POSITA would have understood and found it obvious to implement components shared commonly in both systems (e.g., input devices, monitors) within HAC, accounting for the descriptions provided by both

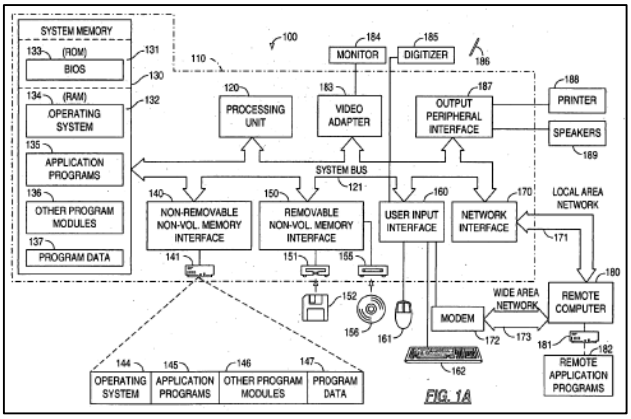
Anthony discloses displaying its windows on a 96 dpi (dots-per-inch) screen, and neither Anthony nor Hanggie teach or suggest that there is a modification in the monitor 191/184's resolution in different operating modes. APPLE1007, ¶[0050]; APPLE1003, ¶[88].

[1a]

Consistent with the teachings of Hanggie and Anthony (e.g., Hanggie's FIG. 1 and Anthony's FIG. 1A shown below), a POSITA would have been motivated and found it obvious to implement HAC such that a plurality of user inputs can be received through input devices such as a keyboard 162, pointing device 161, or stylus, and such that a plurality of applications could be opened in response to receiving the plurality of inputs. APPLE1007, ¶[0038], FIG. 1; APPLE1006, ¶[0005], FIG. 1A, APPLE1003, ¶¶[89]-[91].

references. APPLE1003, pg. 56; APPLE1007, FIG. 1; APPLE1006, FIG. 1A.





APPLE1007, FIG. 1 (left); APPLE1006, FIG. 1A (right).

For example, Hanggie teaches that its CDWM receives requests to create a window and attach a content object. APPLE1006, ¶¶[0050], [0051], [0088]. Although Hanggie does not explicitly teach that the request is a user input, it would have been obvious to a POSITA to implement HAC such that an application window would be generated in response to a user request to open the window received through an input device. APPLE1003, ¶[90]. Moreover, it would have been obvious that a user interface, like the one in HAC, is configured to receive user inputs as receiving user input is what a user interface is deigned to do. APPLE1003, ¶[90].

The above-described method for opening windows, such as application file windows, was well known in the art. APPLE1003, ¶[91]. For example, Anthony teaches that a user may navigate to a particular window and open a file within the window. APPLE1007, ¶¶[0038], [0043]-[0044], [0047] ("[i]f an icon 606a in the focal group 509 represents a file, double-clicking the file will cause the system to

attempt to open the file"), [0084]. And while this disclosure is made with respect to Anthony's 3D timeline, it would have been obvious to a POSITA that a similar operation would have been performed in a 2D environment such that a user would have provided an input to the HAC system when desiring to open a file, such as an application file. APPLE1003, ¶[91].

In addition, it would also have been obvious to a POSITA that when a user opens three application files using one or more input devices, HAC's computer system would have received at least first, second, and third inputs corresponding to the opening of the first, second, and third application files. APPLE1003, ¶[92]. For instance, when a user selects three application files to open using an input device, e.g., by double clicking the mouse when it is positioned over icons corresponding to the three application files, HAC's computer system would receive at least first, second, and third inputs from the mouse. APPLE1003, ¶[92].

[1b]

As noted in [1a], when a user double clicks on a file, HAC's computer system opens it. APPLE1007, ¶[0047]. When the file is an application file, Hanggie's CDWM generates a window having content ("***application-specific data***") corresponding to the application file selected by the user. APPLE1006, Abstract, ¶¶[0005], [0015] ("window content"), [0049]-[0050] ("content object"), [0060]; APPLE1003, ¶[93]. A user may choose to display the window in a legacy

drawing mode, which is 2D, when desired (e.g., to preserve power) by making such a selection via a user control. APPLE1006, ¶¶[0092], [0042]; *see supra* §VI.A.2. Thus, in HAC, the application window ("***object***") is displayed in said 2D space on a fixed resolution display, e.g., 96 dpi, of monitor 191. APPLE1007, ¶[0050], Abstract, FIG. 1; APPLE1003, ¶[93].

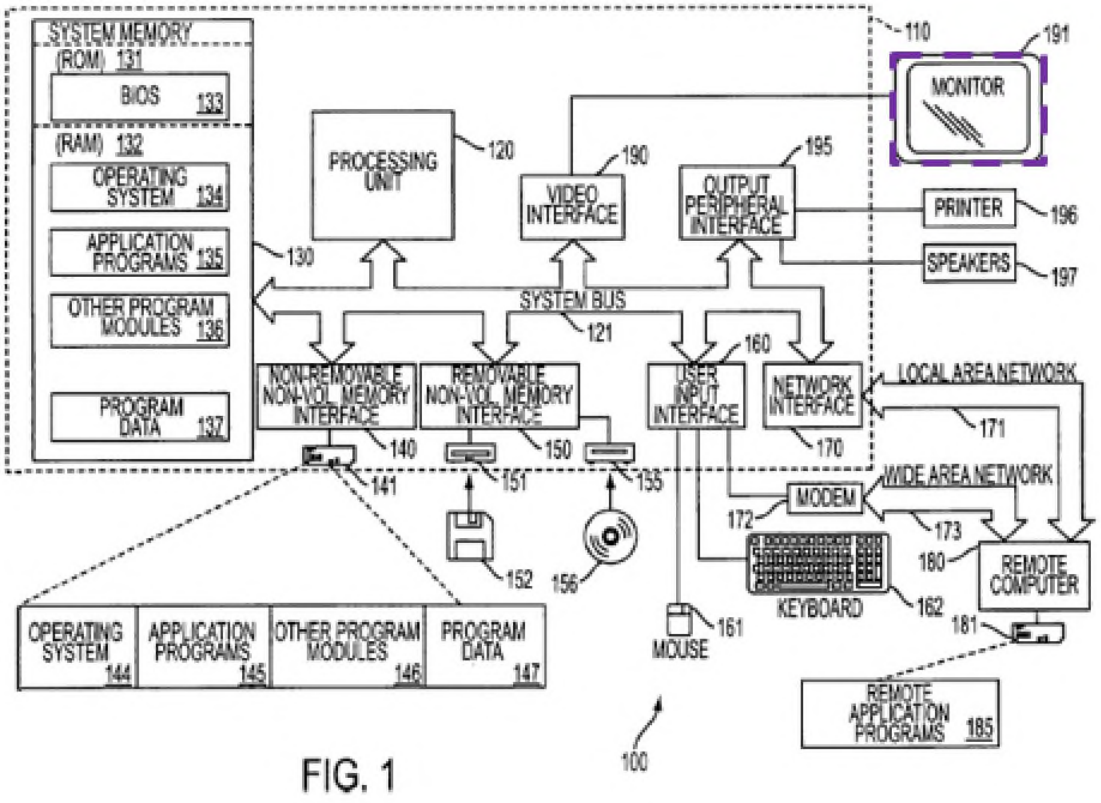


FIG. 1

APPLE1007, FIG. 1.

Hanggie also discloses that 2D graphics can be provided for legacy applications using the Legacy Graphics Device Interface 192b in the Legacy Graphical User Interface subsystem. APPLE1006, ¶[0042]. Content related to the applications can include "e.g., text by a word processor, numeric grid by a

spreadsheet application, or images by a photo editing application." APPLE1006, ¶[0060]. It would have been obvious to a POSITA and well known in the art, that a user can interact with 2D windows to modify at least a portion of the application content ("application-specific data"). APPLE1003, ¶[94]. For instance, when using a word processor application or spreadsheet application, it is well known that a user can interact with the application and edit text and images in the application window using the computer's keyboard, mouse, and/or other input devices. APPLE1003, ¶[94] (citing to APPLE1011, APPLE1012, APPLE1013). Hanggie teaches that "[w]indows can typically be interactively resized, moved around the display, and arranged in stacked order so as to fully or partially overlap one another." APPLE1006, ¶[0003]. A user can provide input for one or more windows using an input device such as a keyboard or mouse. APPLE1006, ¶[0003]; APPLE1003, ¶[94].

Hanggie's FIG. 4 illustrates a method by which windows can be created in a legacy mode using the Legacy Window Manager 192a. Hanggie discloses that "[s]teps 401–409 [in FIG. 4] represent the initial rendering of content associated with a legacy application window whose source rendering surface (or set if [sic] instructions required to generate the surface) is obtained from the Legacy Window Manager 192a." APPLE1006, ¶[0087]. Hanggie explains that "a legacy application call[s] legacy APIs 192b to draw a window on the desktop according to

the invalidation model for which the application was designed. For example, Microsoft® Word® XP may call the legacy APIs so that the legacy DWM 192a draws text input by the user." APPLE1006, ¶[0088]. Operations 403-409 are then performed to render the application window with the desired content, properties, and parameters. APPLE1006, ¶[0088]; APPLE1003, ¶[95].

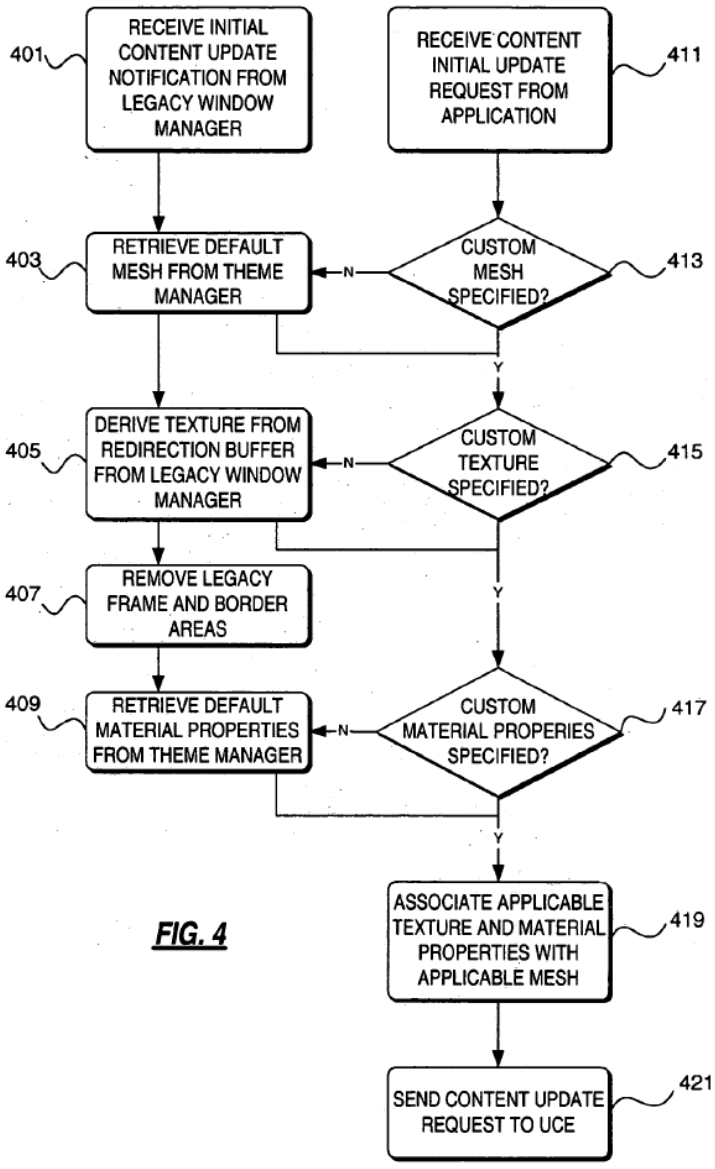


FIG. 4

APPLE1006, FIG. 4.

[1c]

As explained in [1b], a user can interact with the 2D windows to switch between a plurality of applications in the HAC system. APPLE1006, ¶[0003] ("Windows can typically be interactively resized, moved around the display, and arranged in stacked order so as to fully or partially overlap one another.") Anthony also discloses that a user can "freely navigate documents, files, or other data objects in a chronological manner," and that the user can "change the point of focus to an arbitrary location on a timeline." APPLE1007, ¶[0044]. As explained in more detail below in §§[1c-1]-[1c-5], HAC renders obvious allowing a user to switch between applications by implementing features [1c-1]-[1c-5]. APPLE1003, ¶[96].

[1c-1]

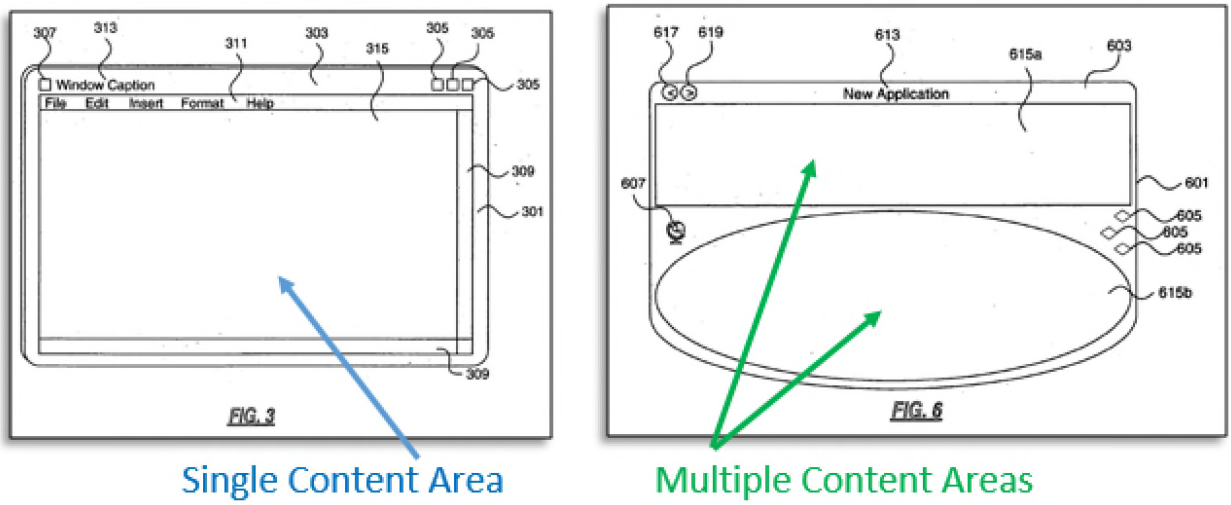
In HAC's system, a user can view data in 2D and 3D (e.g., in Anthony's 3D timeline or a 2D window). APPLE1007, FIGS. 4, 5; APPLE1006, ¶¶[0041]-[0044], [0056], [0065]. In particular, as explained above in §VI.A.2 and [1b], Hanggie discloses that a user can select whether to operate the computer in DWM (legacy mode with 2D display of data) mode or in CDWM (3D display of data). APPLE1006, ¶¶[0092], [0042]. Operations 401-409 in FIG. 4 describe how a 2D window is rendered in the DWM mode. *See supra* [1b]; APPLE1006, ¶¶[0087], [0088], FIG. 4. It would have been obvious to a POSITA that when a user views

the three 2D windows (as described above for [1b]) in the DWM mode and then selects the CDWM mode to interact and view the windows in 3D, HAC's computer would ***generate a plurality of images in the form of 3D windows that respectively correspond to the three 2D windows***. APPLE1003, ¶[97]; APPLE1006, ¶¶[0026], [0057] ("to draw the window in 3D, the CDWM may define the window anatomy using various components), [0058], [0059] ("[t]he CDWM may support multiple application created and rendered content areas, associated with a single window"); FIG. 3. Specifically, Hanggie discloses that an application can generate an application window and attach any number of content objects to the application window. APPLE1006, ¶¶[0049]-[0050], [0087]-[0092], FIGS. 2, 4. In generating a 3D composited window, the computer modifies various features of the (previous) 2D window such as the "content size, window position or scale, or a change to the pixels of the contents diffuse texture." APPLE1006, ¶[0050]. "[T]he CDWM derives composited window attributes (frame and border style, caption, etc) and state (hidden/shown, minimized/maximized, etc) from existing legacy window properties" and "the legacy window manager 192a informs the CDWM 190 of any change affecting the redirected window content texture that may necessitate a visual update." APPLE1006, ¶[0051]; APPLE1003, ¶[97].

Hanggie, like Anthony, teaches that a 3D graphics engine such as Microsoft Direct3D® can be used to render 3D objects. APPLE1006, ¶[0056]; APPLE1007,

¶[0043]. Hanggie explains that "[i]n order to draw the window in 3D, the CDWM may define the window anatomy using various components, include a base content object and one or more child content objects. The base content object defines the window frame, or border, and consists of a base geometry, base extent, base material properties and base content margins. The base and child content objects may each be entirely defined and managed by the system or in the case of custom content elements, may be managed by the application." APPLE1006, ¶[0057]. Like Anthony, Hanggie, explains that multiple windows, applications, and frame contents can be displayed, removed, and repositioned, and that windows can be defined by an application. APPLE1006, ¶¶[0058]-[0060]; APPLE1003, ¶[98].

As an example of multiple windows created for respective applications, compare Hanggie's FIG. 3, which illustrates a window generated for one application, with Hanggie's FIG. 6, which illustrates a window generated for a second application. APPLE1006, ¶¶[0058], [0064]. The application window associated with FIG. 3 has a single object area 315 whereas the application window associated with FIG. 6 has "a dynamic non-standard anatomy" of two object areas 615a, 615b. APPLE1006, ¶¶[0058], [0064]-[0065]; APPLE1003, ¶[99].



APPLE1006, FIG. 3(left), FIG. 6 (right).

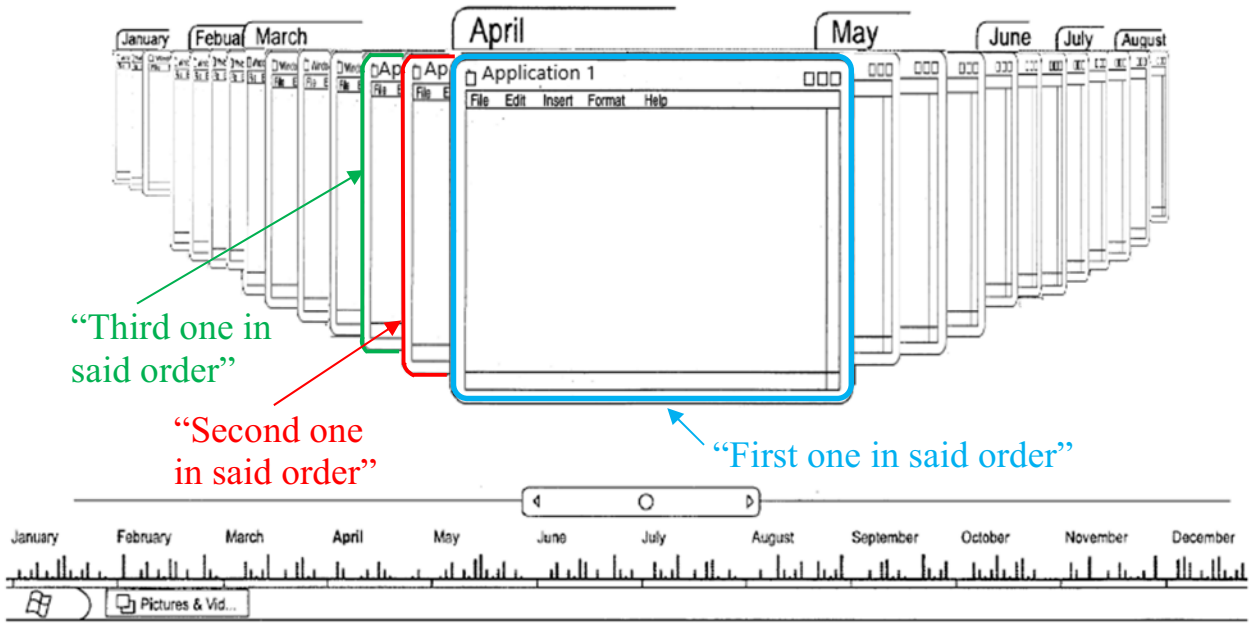
In the above-described manner, it would have been obvious to a POSITA that for each of the three 3D application windows that is generated and displayed, the respective first, second, or third application window is an image of at least a portion of an object (e.g., 2D window) generated by said the respective application and having respective application-specific data (e.g., application data, application content object). APPLE1003, ¶[100].

[1c-2]

HAC renders obvious "replacing all objects corresponding to said plurality of applications that are visible in said 2D space with said plurality of images" for at least the reasons noted above in [1c-1]. *See supra* [1c-1]; APPLE1003, ¶[101].

Anthony explains that the multiple windows can be arranged according to a particular "ordering attribute," which may be based on chronologically in time "e.g., by using a date of creation or date of edit attribute as the ordering attribute."

APPLE1007, ¶[0042]. It would have been obvious to a POSITA that the last date of edit of a window, in some cases, corresponds to the last time a user opened *and* interacted with the window associated with an application. APPLE1003, ¶[102]. Accordingly, each of HAC's 3D windows ("***plurality of images***") are displayed in an order based on a last time that the user opened the application corresponding to a window *and* interacted with the window. APPLE1003, ¶[102]. As shown in the combination figure below, the three windows can be displayed in an order from the foreground to behind the background such that "a first one in said order is displayed in a foreground of said 3D immersive space, a second one in said order is displayed in a background of said 3D immersive space behind at least said first one in said order, and a third one in said order is displayed in said background of said 3D immersive space behind at least said second one in said order." APPLE1003, ¶[102].



APPLE1003, pg. 68.

[1c-3]

Anthony teaches that a user can freely navigate through the files, folders, and objects in the timeline to focus on any point in the timeline. APPLE1007, ¶[0044]. To facilitate user navigation, Anthony discloses a scroll bar or wheel to shift to different points of the timeline. APPLE1007, ¶¶[0081], [0084]. When a user manipulates the timeline to change the focus onto a different window or focal group, all the windows in the timeline shift so that the window that the user wants to view is at the focal point displayed in front of the user's view. APPLE1007, ¶[0044], FIG. 5; APPLE1003, ¶[103].

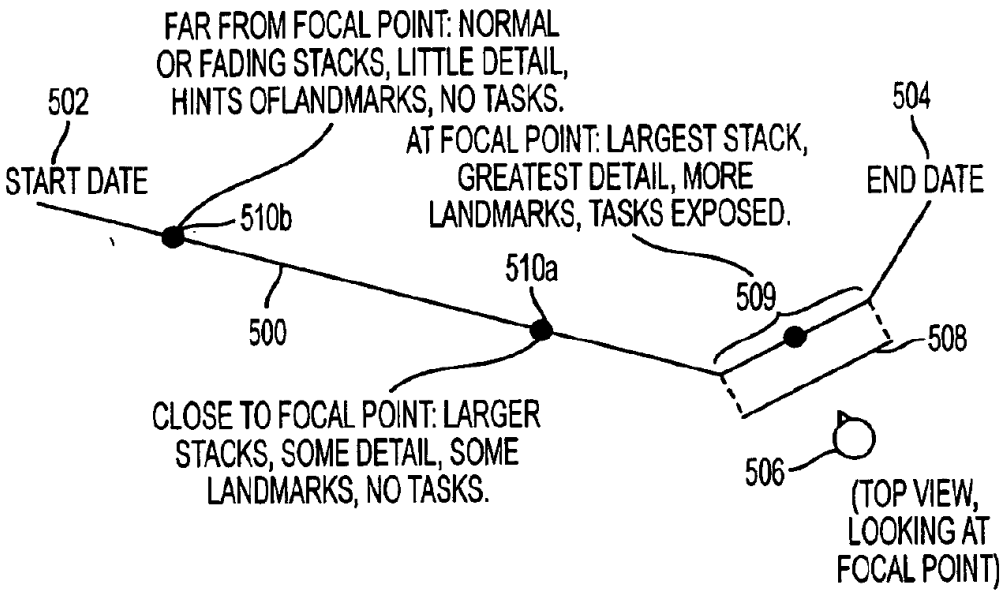
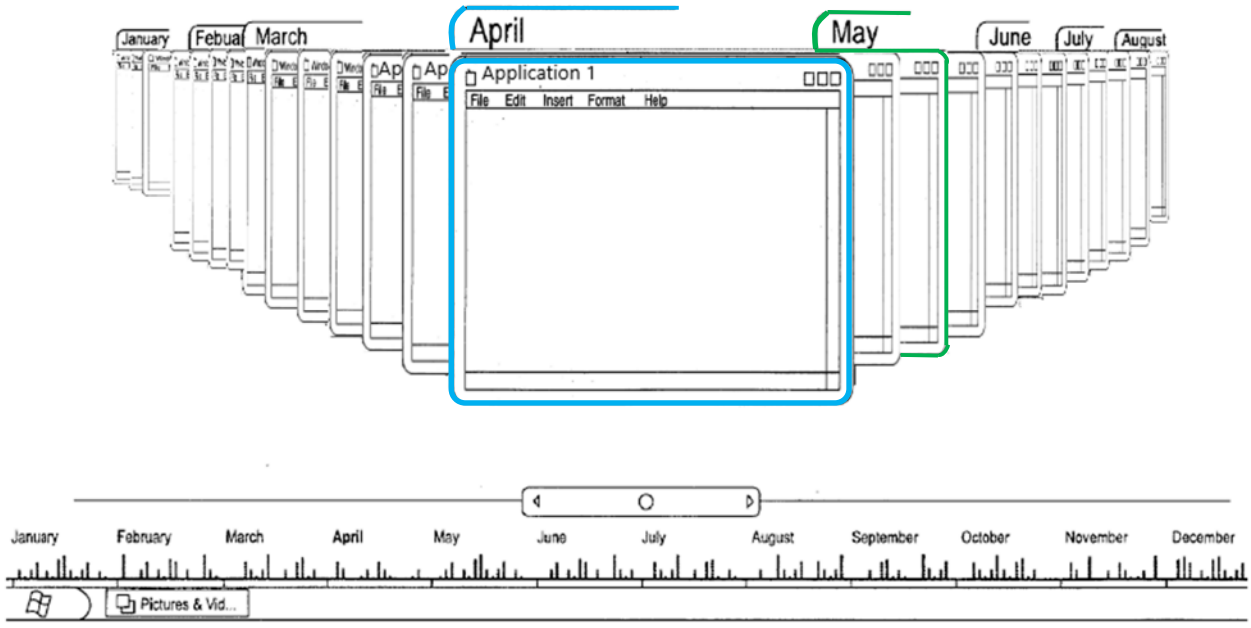


FIG. 5

APPLE1007, FIG. 5.

For example, referring to the combination figure below, when a user wants to shift from viewing a window in April to a window in May, all the windows shift to the left such that the May window now occupies the center of the screen instead of the April window. June and latter months shift closer to the center of the screen and April and earlier months shift away from the center of the screen.

APPLE1003, ¶[104].



APPLE1003, pg. 70.

[1c-4]

Anthony refers to the windows that are not displayed at the focal point as non-focal groups. APPLE1007, FIGS. 5, 6, ¶¶[0044], [0065]-[0066]. Anthony explains that "as the non-focal groups are set farther and farther away from the focal group 509, the non-focal groups may become blurred, fogged or transparent. If the number of intervals between a non-focal group and the focal group becomes large, the non-focal group might not be displayed, or it may simply become too small for the user to see on the screen." APPLE1007, ¶[0066]. Referring to Anthony's FIG. 11, some windows are ***continuously moved in one direction*** so far from the focal point that they no longer appear in the visible area of the timeline and ***result in a perception that the windows have been moved off the display***

screen, ***which has a fixed resolution***. APPLE1007, ¶[0063], FIG. 11; APPLE1003, ¶[105].

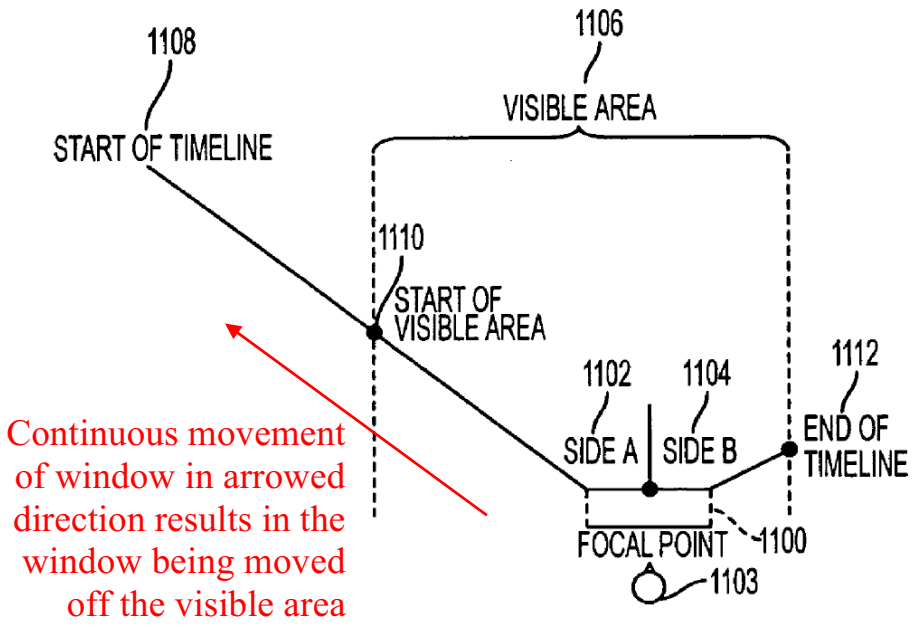


FIG. 11

APPLE1007, FIG. 11.

[1c-5]

As explained above in §§VI.A.1-VI.A.4 and shown in the figures below, in HAC's system, HAC's 3D windows include Hanggie's buttons 305 that can be used to "used to restore, maximize, minimize, close the window, etc." APPLE1006, ¶[0058]. It would have been obvious to a POSITA that a user would have been allowed "***to delete at least one of said plurality of images from said 3D immersive space***" by selecting a button that closes the window. APPLE1003, ¶[106]. It is known in the art that closing a window results in deleting the window

from display. APPLE1003, ¶[106].

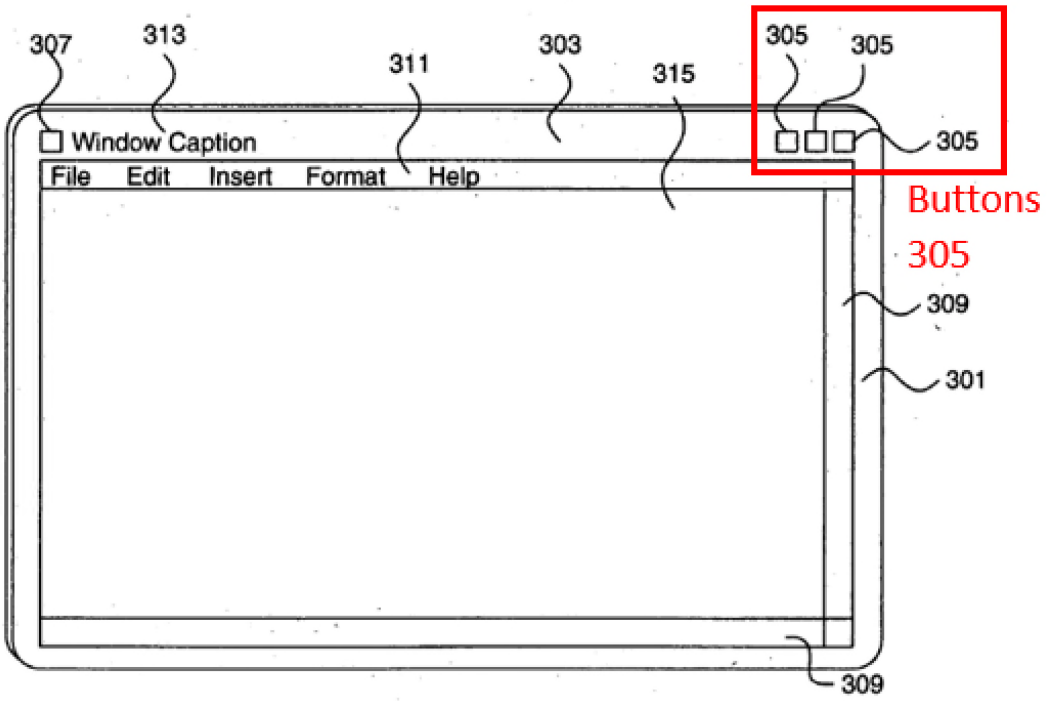
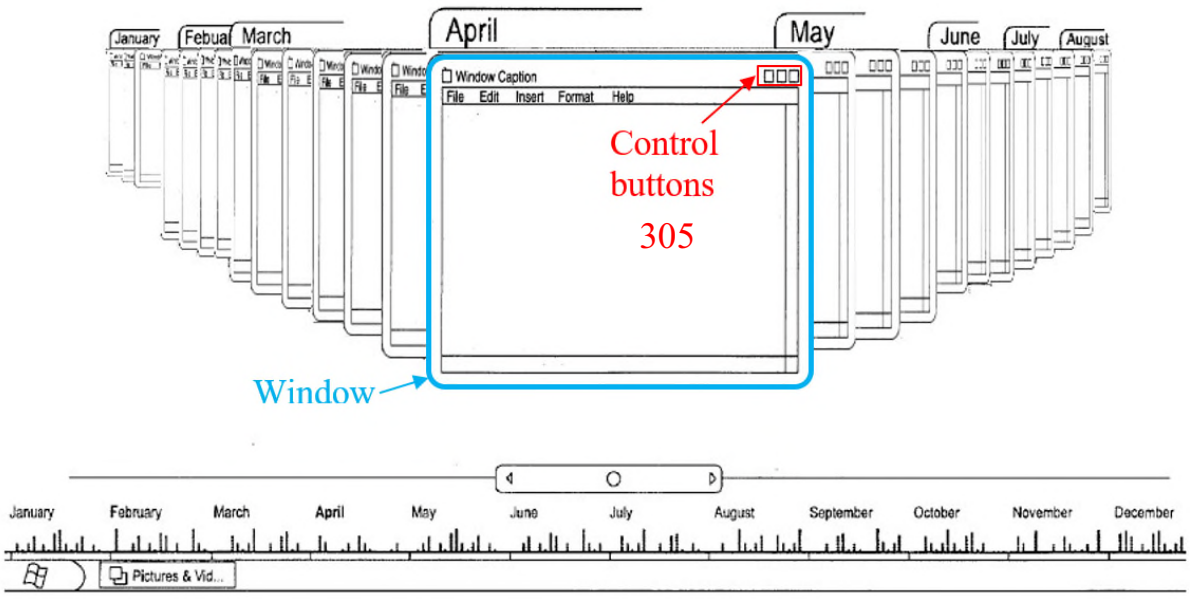


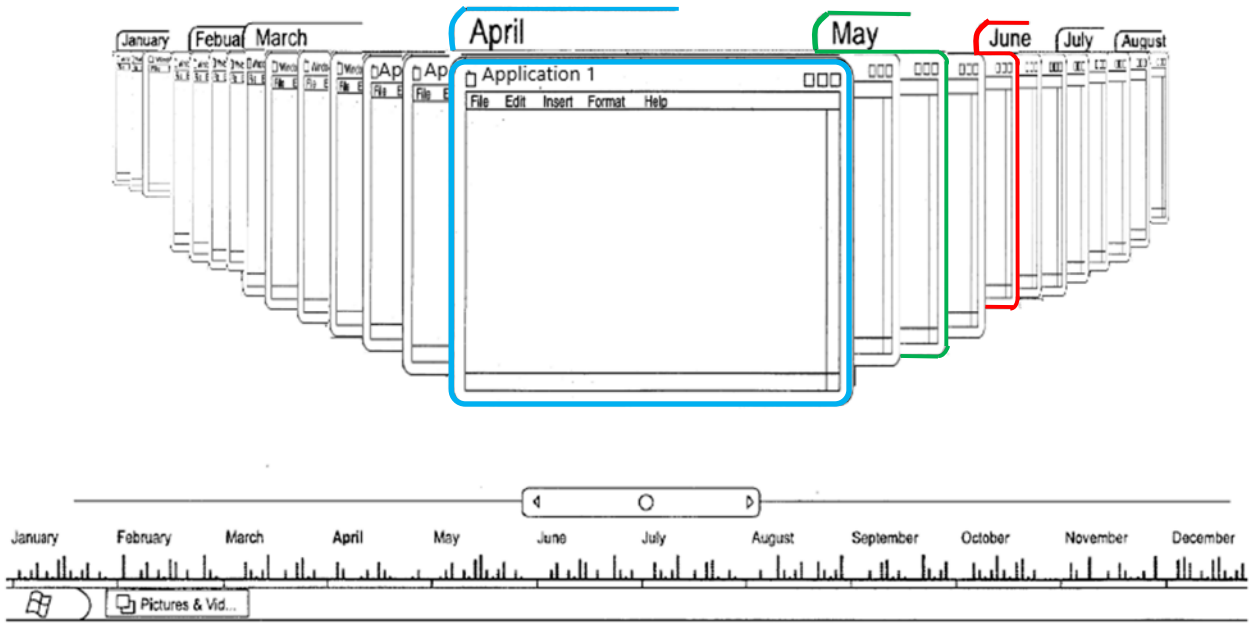
FIG. 3

APPLE1006, FIG. 3



APPLE1003, pg. 73.

When such a deletion is performed in Anthony's dynamic timeline, it would have been obvious that other windows behind the deleted window would move towards the focal point. APPLE1003, ¶[107]. For example, referring again to the Anthony-Hanggie combination figure below, when a window (e.g., May window) behind and to the right of the April window (focal point) is deleted, a window (e.g., a June window) behind the deleted window (deleted May window) would then move closer to the focal point at the location previously occupied by the deleted window. APPLE1003, ¶[107].



APPLE1003, pg. 74.

[1d]

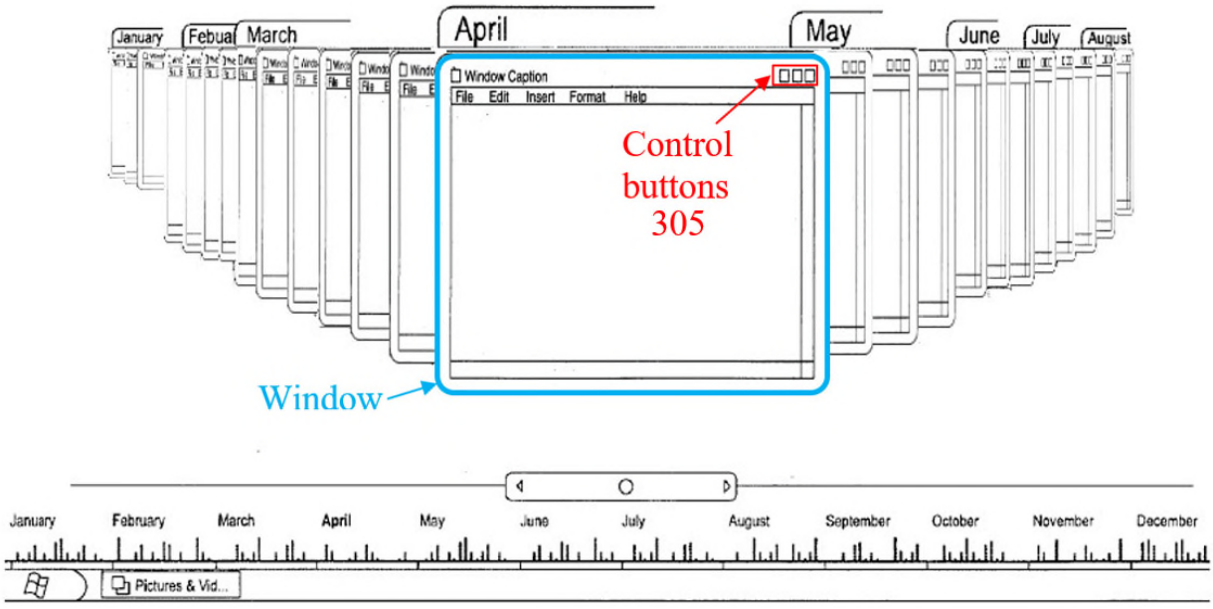
HAC discloses several methods through which a user can interact with any one of the displayed application windows corresponding to various applications.

For example, as explained in [1c-3]-[1c-5], the windows can be moved, deleted, or minimized. HAC also renders obvious the interactions claimed in [1d-1]-[1d-4] as explained in detail below. APPLE1003, ¶[108].

[1d-1]

Recall that Hanggie discloses that a user can make a selection to operate in a 2D (DWM) or 3D (CDWM) mode. *See supra* [1pre], [1b]; APPLE1006, ¶[0092]. Hanggie explains that, in some implementations, the selection can be made manually or "in accordance with the drawing mode defined by an activated visual style (theme) selected by the user." APPLE1006, ¶[0092]; APPLE1003, ¶[109].

Recall from [1c-2] that HAC's 3D windows include buttons 305 that can be used to "used to restore, maximize, minimize, close the window, etc." APPLE1006, ¶[0058], FIG. 3; APPLE1003, ¶[110].



APPLE1003, pg. 75.

Upon ***receiving a selection from a user*** of the maximize button for a window ("***of one of said plurality of images corresponding to one of said plurality of applications***"), it would have been obvious to a POSITA that HAC's computer would have allowed a user to view the window in a "maximum" capacity such that it occupies all or almost all parts of the computer display. APPLE1003, ¶[111]. And because the window occupies most, if not all, or the display screen, viewing the window in 2D would allow a user to more intuitively and easily interact with the window. APPLE1003, ¶[111]. Accordingly, using the maximize button to select 2D viewing of a window would have been an obvious choice to provide a manual selection option for a displayed window to be viewed in 2D. APPLE1003, ¶[111]; APPLE1006, ¶[0092]. Moreover, because the experience of

viewing a maximized window in 2D or 3D would not have provided any substantial or perceived difference in viewing experience, using the maximize button to switch to a 2D viewing of the window would have been obvious. APPLE1003, ¶[111]. In this manner, by selecting a maximize button on a window in Anthony's timeline, the window would be reconfigured in the legacy mode and displayed in 2D to occupy all or most parts of the screen in a "maximum" capacity. Such a maximized 2D window would then replace the display of the dynamic timeline and any windows within the timeline. APPLE1003, ¶[111].

Indeed, using buttons such as the maximize button is consistent with the manner in which the '868 Patent describes switching viewpoints. APPLE1003, ¶[112]. In particular, the '868 Patent explains that by "clicking an icon or bottom [sic] (analogous to the minimize in windows operating system environment) triggers a change to the viewpoint of the end user" from 3D to 2D. APPLE1001, 22:7-30.

[1d-2]

See supra [1d-1]; APPLE1003, ¶[113].

[1d-3]

Hanggie discloses several way in which the computer receives at least one user interaction with a 2D window, such as a maximized 2D window. For example, if the maximized window was a Microsoft® Word® application window,

Hanggie discloses receiving text input from a user. APPLE1006, ¶[0088]. As is known in the art, in response to a user entering a key on a keyboard when interacting with a Microsoft® Word® application window, application-specific data, such as the text content in the window is modified to reflect the input key. APPLE1003, ¶[114]. It would also have been obvious to a POSITA that when a user would like to interact with multiple 2D windows, the computer would have received multiple user interactions for the user to interact with the multiple windows (e.g., second and third objects) in the 2D space. APPLE1003, ¶[114]. Subsequently, application-specific data in such windows would have been modified similar to the single window example. APPLE1003, ¶[114].

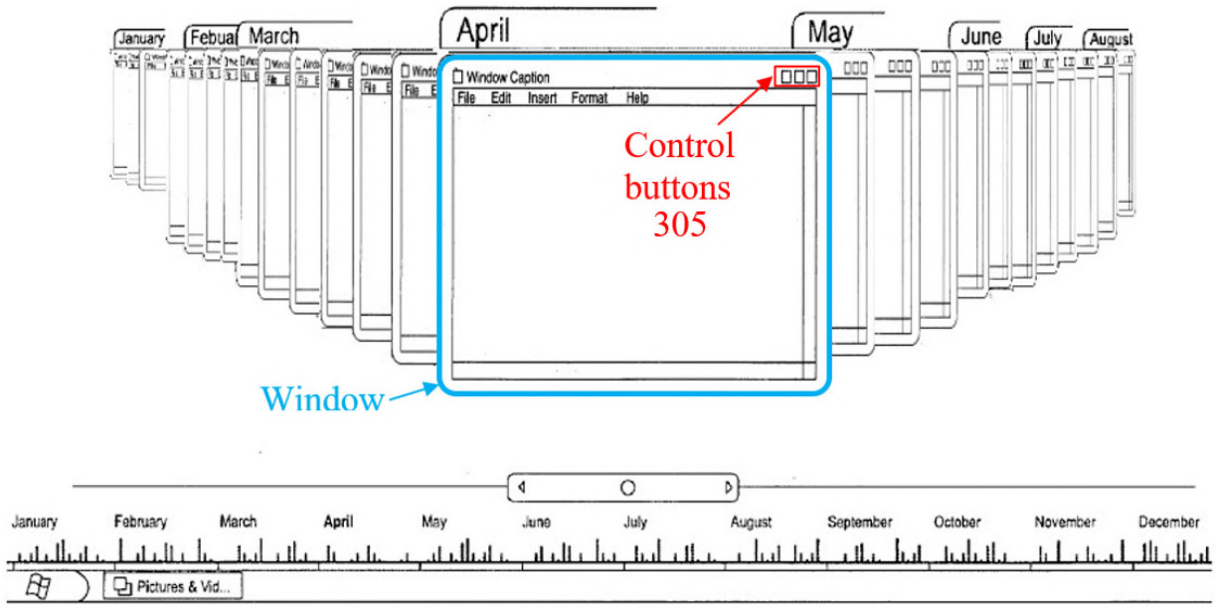
[1d-4]

See supra [1d-3]; APPLE1003, ¶[115].

Claim 2 - ***[2]***

As noted above in §§VI.A.1-VI.A.4 and [1c-1], Hanggie teaches that "[t]he frame or base content 303 of the window 301 may host child content including **buttons 305 (e.g., used to restore, maximize, minimize, close the window, etc.),** an indicative icon 307, scrollbars 309, menu bar 311, and window caption text 313." APPLE1006, ¶[0058], FIG. 3. ***Any one of the buttons 305 is an icon*** that is displayed together with each window ("***image***"). Accordingly, it would have been

obvious to a POSITA that when a plurality of windows ("***images***") is displayed in HAC, the plurality of windows is displayed together with a plurality of icons, and each of the windows has a corresponding one of the plurality of icons, as shown in the combined HAC figure below. APPLE1003, ¶[116].



APPLE1003, pg. 79.

Claim 3 - ***[3]***

As noted above in §§VI.A.1-VI.A.4 and [1c-1], Hanggie teaches that "[t]he frame or base content 303 of the window 301 may host child content including buttons 305 (e.g., used to restore, maximize, minimize, close the window, etc.), an **indicative icon 307,** scrollbars 309, menu bar 311, and **window caption text 313.**" APPLE1006, ¶[0058]. As shown in Hanggie's FIG. 3 (reproduced below), each window ("***image***") has a corresponding indicative icon 307 and window caption

text 313 ("***descriptions***"). It would have been obvious to a POSITA that when a plurality of windows ("***plurality of images***") are displayed, each one of said plurality of windows has a corresponding one of said plurality of indicative icons 307 and window caption text 313 ("***plurality of descriptions***"). APPLE1003, ¶[117].

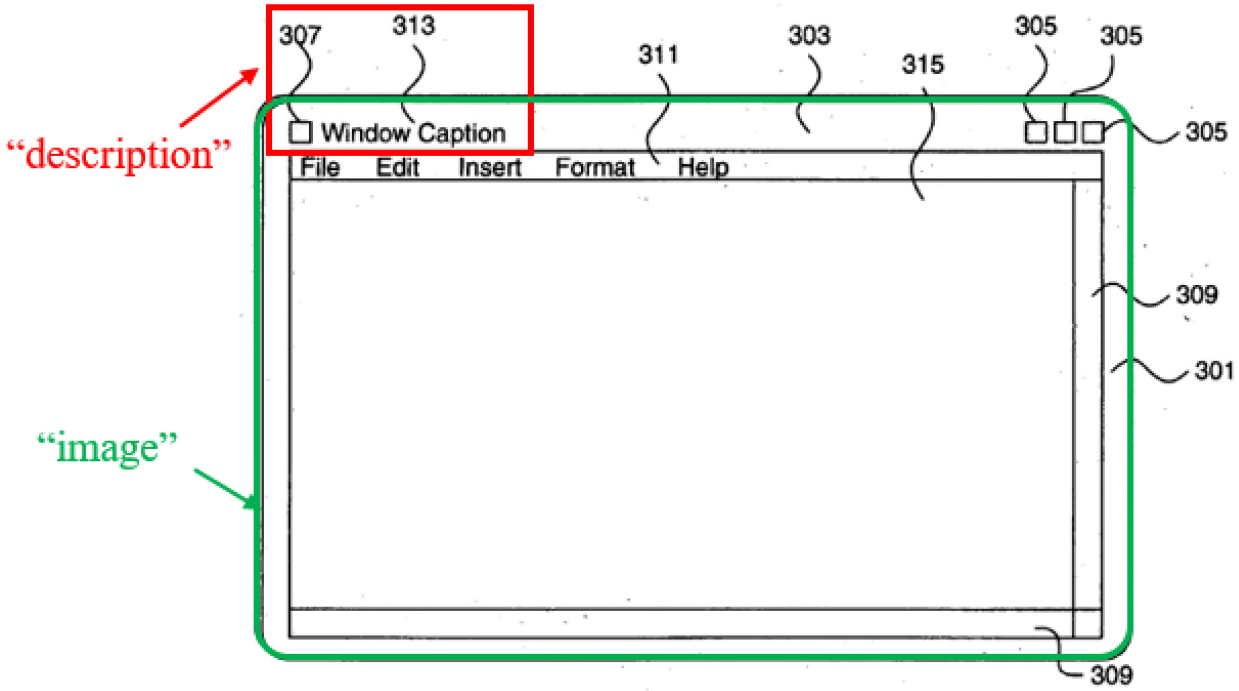


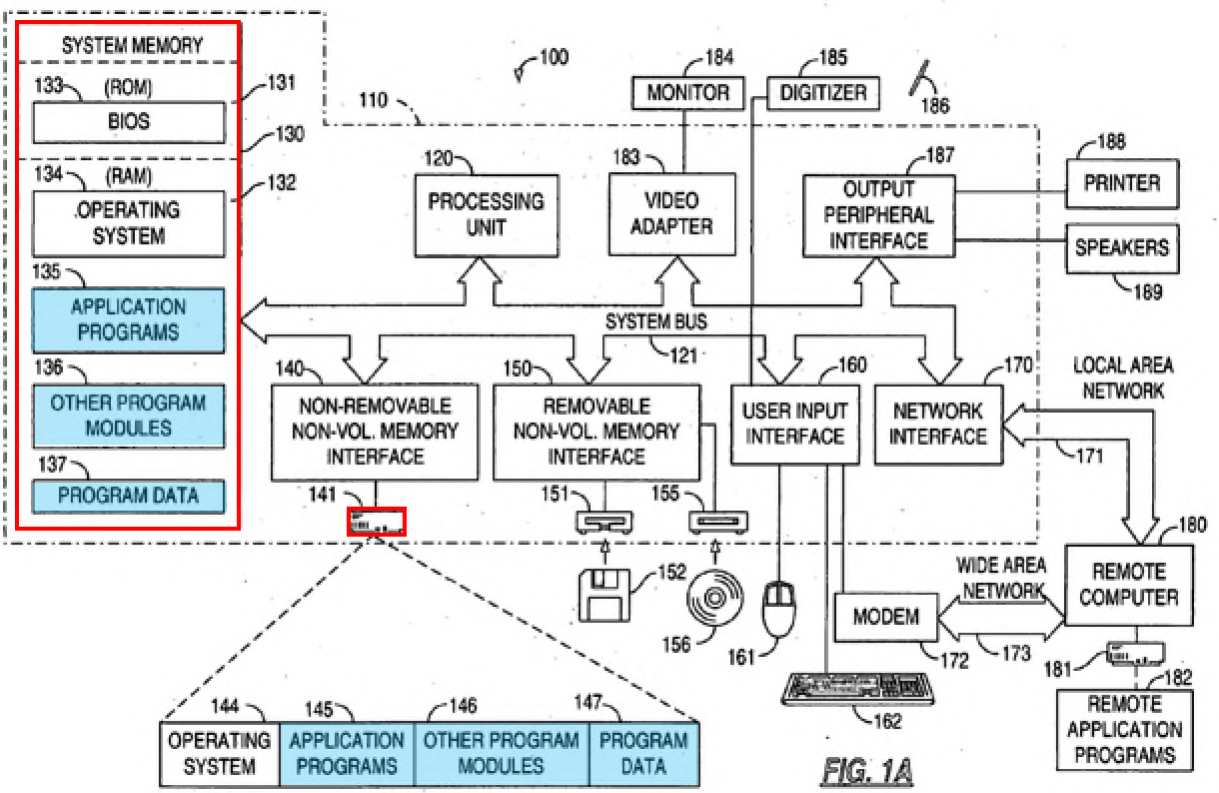
FIG. 3

APPLE1006, FIG. 3.

*Claim 4 - **[4]***

As shown in Hanggie's FIG. 1A, Hanggie teaches that includes computer includes a system memory 130 and hard disk drive 141 that include application programs 135/145, program modules 136/146, and program data 137/147.

APPLE1006, ¶¶[0033]-[0035]; APPLE1003, ¶[118].



APPLE1006, FIG. 1A.

The stored application programs and program data are used to configure the displayed windows (images). APPLE1006, ¶¶[0049]-[0052], FIG. 2. For example, Hanggie teaches that "an application can dynamically create and attach to a window (as well as detach and destroy) any number of content objects." APPLE1006, ¶[0049]. Rendering features (e.g., transparency, size, orientation) of the window or any changes to the window or content therein can also be controlled by an application program. *Id.*, ¶¶[0052], [0054], [0059], [0060]; APPLE1003, ¶[119]. Hanggie discloses that:

> basic window elements, and that additional or different window elements may additionally or alternatively be used … [and] may alternatively be provided by an application, e.g., to provide a distinct look and feel to an application program. An example would be where an application program provides the scroll bar elements as custom child content objects so that they manifest an appearance and behavior peculiar to the application program. Moreover, an application may elect to remove or reposition one or more of the stock frame elements using the CDWM API. An application need not be limited to a single primary content area, a restriction prevalent in the prior art.

APPLE1006, ¶[0058].

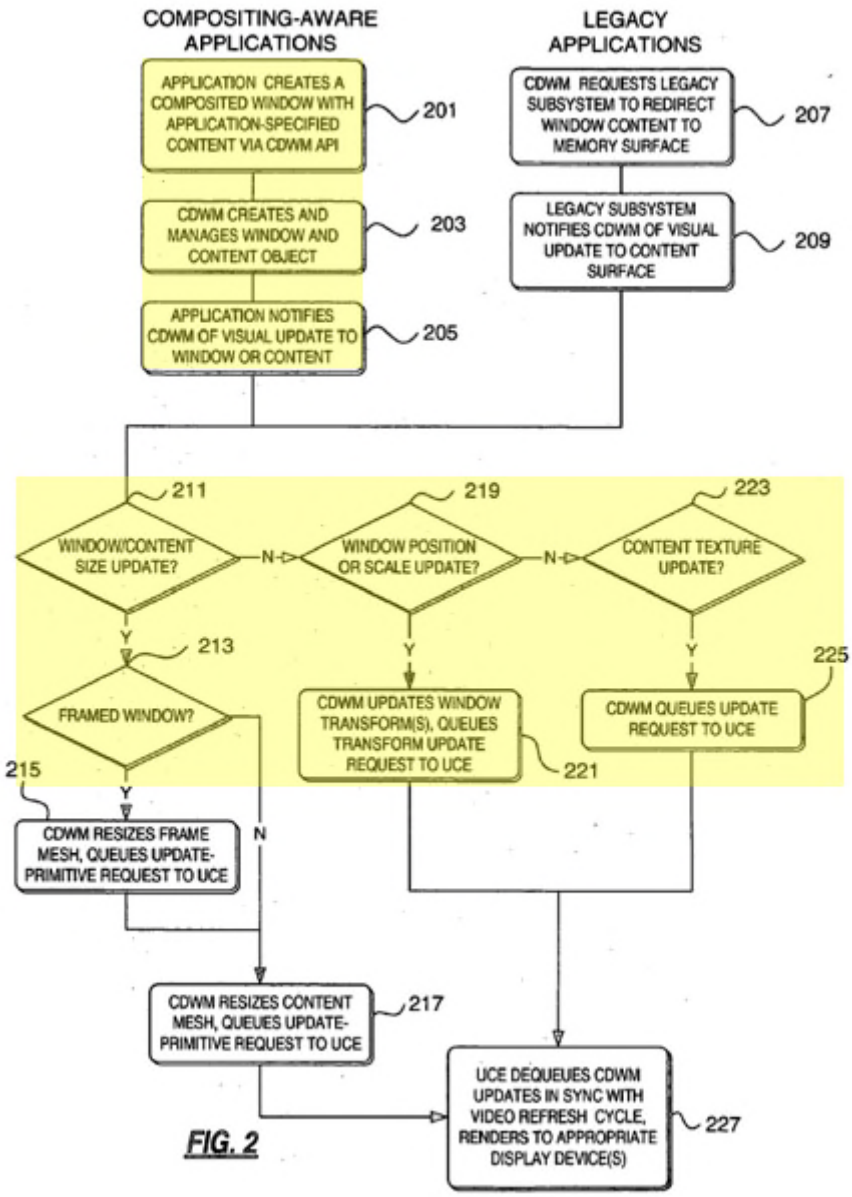


FIG. 2

APPLE1006, FIG. 2.

As can be appreciated from the foregoing, the RAM 132 or hard disk drive 141 ("***database***") ***store data***, such as the application programs 135/145 (optionally including program modules 136/146, and program 137) which are ***linked to each one of the*** displayed windows ("***plurality of applications***") that a respective

application program controls. APPLE1003, ¶[120]. It would have been obvious to a POSITA that when a window corresponding to a particular application program is opened, the application program would have stored data linking the application to the window so that the application program can control and track the content and features of the window. APPLE1003, ¶[120].

Note that "database" is defined as "a nonredundant collection of interrelated data items that can be shared and used by several different subsystems." APPLE1010, 136. Accordingly, a POSITA would have understood that HAC's RAM 132 or hard disk drive 141 is a database because RAM 132 or hard disk drive 141 store data that is shared and used by several different subsystems in HAC's system such as the processing unit 120 and monitor 184. APPLE1003, ¶[121].

Recall that Anthony discloses an ordering attribute based on time, e.g., by using a date of creation or date of edit attribute as the ordering attribute. APPLE1007, ¶[0042]. When displaying multiple windows, it would have been obvious to a POSITA that HAC's computer determines and stores the ordering information providing by the ordering attribute to arrange and display the multiple windows. APPLE1003, ¶[122]. This order information would have been recalled from a database to support graphical rendering of the windows, and would have been dynamically updated as the ordering of windows changed over time based on

user interaction with the windows. APPLE1003, ¶[122]. Accordingly, it would also have been obvious to a POSITA that HAC's RAM 132 or hard disk drive 141 ("***database***") would also store data for "***a corresponding one of said order in which said plurality of images are displayed on said display device***." APPLE1003, ¶[122].

Claim 5

[5]

Anthony teaches that "[a]ccording to an aspect of the invention, the point of view of the GUI may be based on a conceptual camera 506 pointed at the time interval containing the files or objects on which the user wishes to focus." APPLE1007, FIG. 5, ¶[0044]; APPLE1003, ¶[123]. In describing the time interval, Anthony discloses:

> This time range is called the focal point 508 or focal point interval. The focal point 508 may represent a point in time or an interval of time on the timeline 500. Items that are within the time range covered by the focal point are collectively called the focal group 509. Items in the focal group 509 are typically presented to the user in greater detail than is shown for items in other time intervals on the timeline 500. … **Items in non-focal groups relatively farther from the focal group 509 may**

be shown with very little detail, and may be smaller relative to the items in both those non-focal groups 510a that are closer in time to the focal point 508, as well as the focal group 509 itself. The smaller appearance may be produced automatically when a 3D rendering technology is used, or may be simulated in 2D by drawing the items smaller.

APPLE1007, ¶[0044].

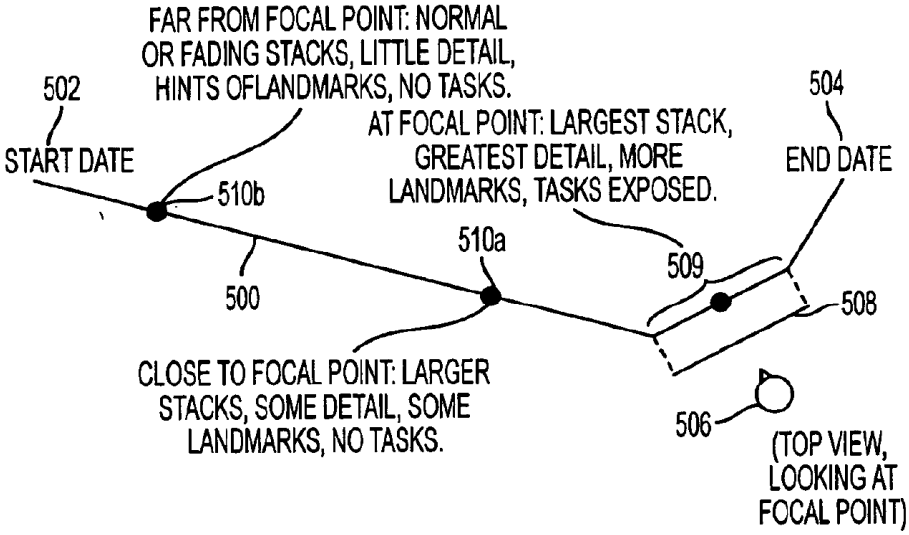
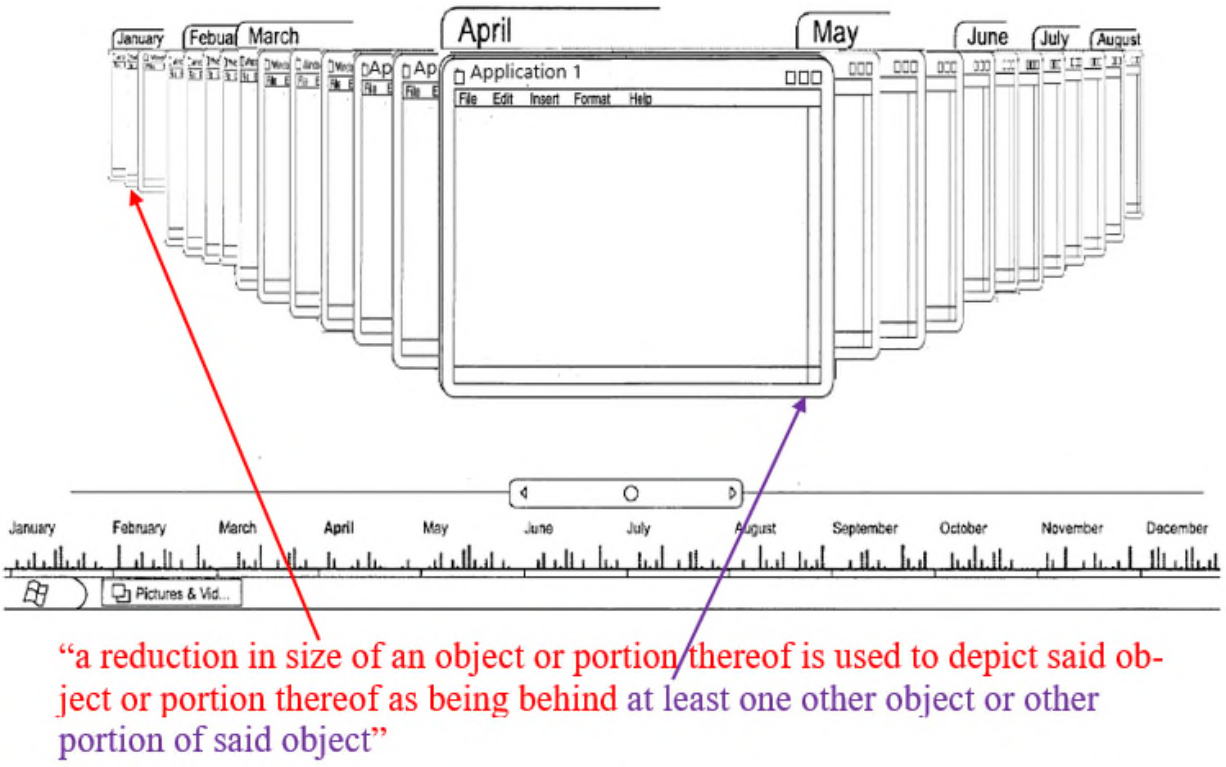


FIG. 5

APPLE1007, FIG. 5.

As can be seen in Anthony's FIG. 5 (reproduced above) and the HAC combination figure (reproduced below), in HAC's system, windows that are further away from the focal point are reduced in size to depict that they are behind the focal point. In this manner, HAC renders [5] obvious. APPLE1003, ¶[124].



APPLE1003, pg. 87.

Claim 9 - ***[9]***

As explained in [1d-1], a maximize button is included on each window (image) corresponding to the first, second, and third applications. It would have been obvious to a POSITA that maximizing a window involves enlarging the size of the window (image) until it covers most if not all the display area of the display screen. APPLE1003, ¶[134]. In gradually increasing its size to cover the display area, the computer simulates movement of said one (window whose maximize button has been selected) of said plurality of images in a z-axis of said display device. APPLE1003, ¶[134].

Claim 10

[10pre]

[10pre] is a system claim preamble corresponding to [1a] (a method claim preamble). HAC renders obvious [10pre] for the same or similar reasons noted above in [1pre]. *See also* §§VI.A.1-VI.A.4; APPLE1003, ¶[135].

[10a]

HAC's system includes a computer 110 connected to a monitor 191/184 ("***display device***"). APPLE1007, FIG. 1, ¶¶[0034], [0038], [0039]; APPLE1006, ¶[0035], FIG. 1A; APPLE1003, ¶[136].

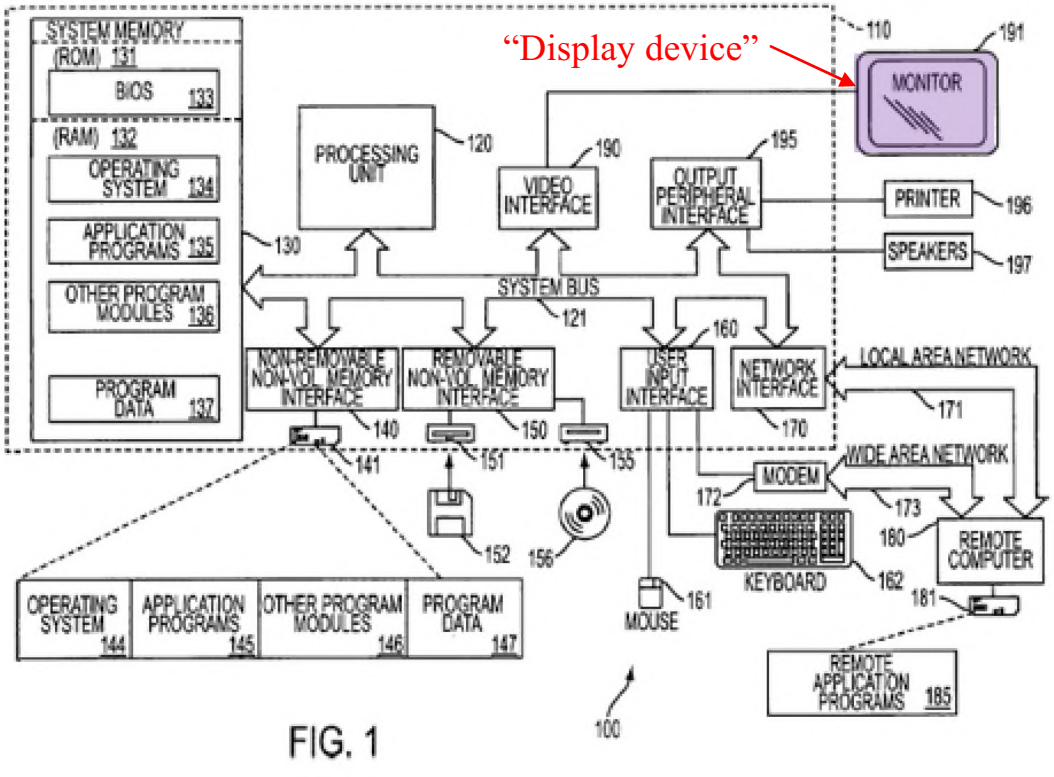


FIG. 1

APPLE1007, FIG. 1.

[10b]

HAC's system includes at least one input device such as "a keyboard 162 and pointing device 161, commonly referred to as a mouse, trackball or touch pad. Other input devices (not shown) may include a microphone, joystick, game pad, satellite dish, scanner, or the like." APPLE1007, FIG. 1, ¶¶[0038], [0034], [0039]; APPLE1006, ¶¶[0005], [0035], FIG. 1A; APPLE1003, ¶[137].

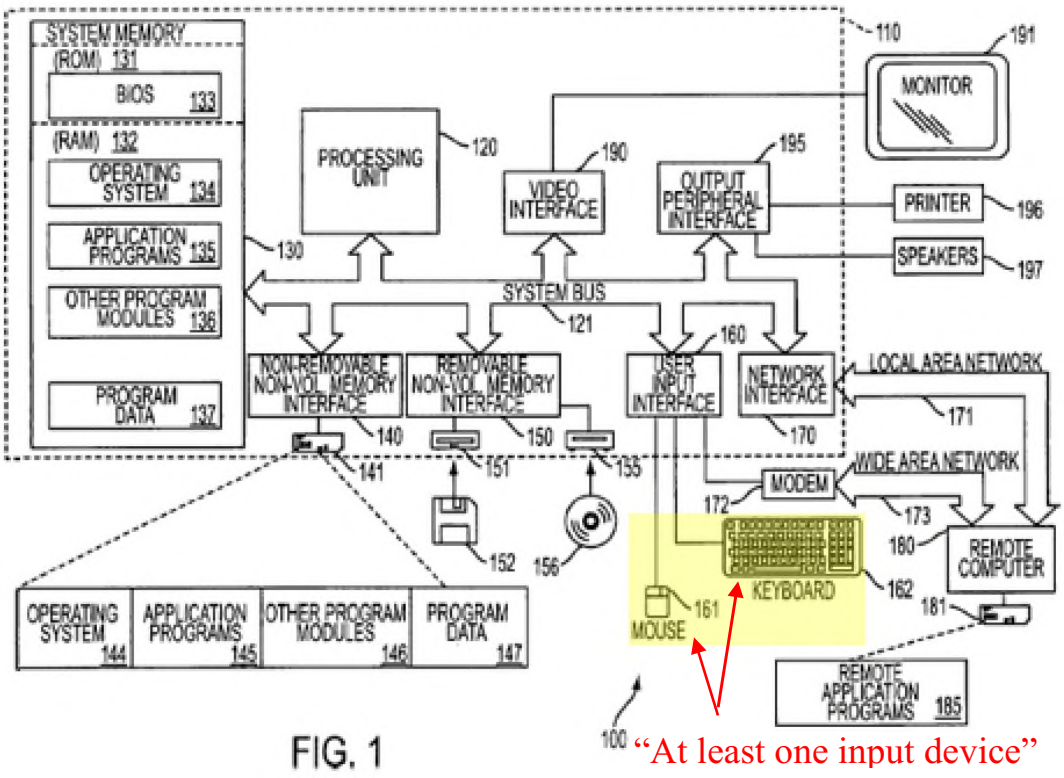


FIG. 1

APPLE1007, FIG. 1.

[10c]

HAC discloses that "[c]omponents of computer 110 may include, but are not limited to, a ***processing unit 120***, a system memory 130, and a system bus 121 that ***couples various system components including the system memory to the***

processing unit 120." APPLE1007, FIG. 1 (reproduced below), ¶[0034]; APPLE1006, ¶[0031], FIG. 1A. More specifically, HAC's system includes a processing unit 120 ("***at least one processor***") operatively coupled to monitor 191 ("***said display device***") and mouse 161, and keyboard 162 ("***said at least one input device***"). APPLE1007, FIG. 1, ¶¶[0038], [0034], [0039]; APPLE1006, ¶[0031], FIG. 1A; APPLE1003, ¶[138].

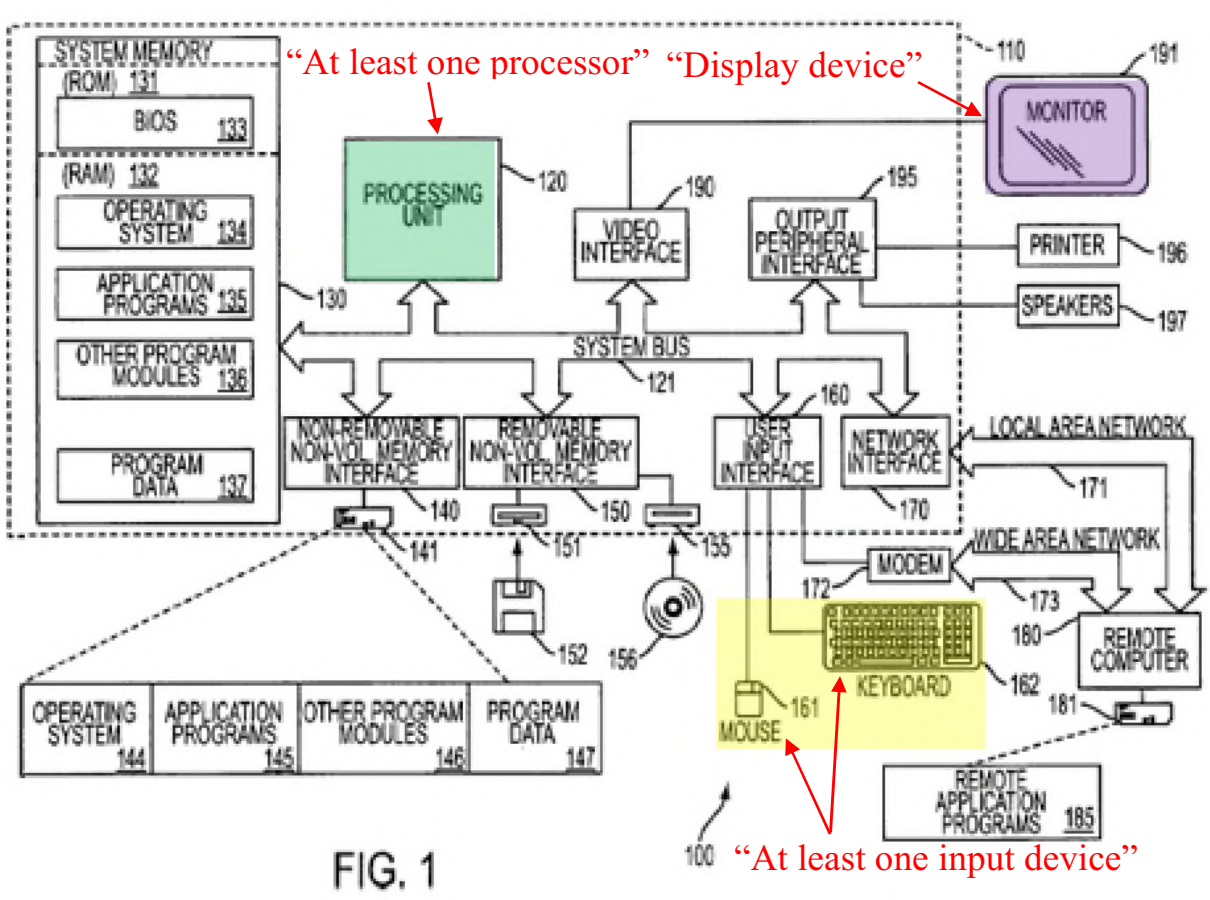


FIG. 1

APPLE1007, FIG. 1.

[10d]

HAC's system includes a memory device, such as system memory 130 that executes code for performing the operations recited in [10e-1]-[10i-4]. APPLE1007, FIG. 1, ¶¶[0034]-[0039]; APPLE1006, ¶[0031], FIG. 1A. For example, Anthony discloses that system memory 130 includes (i) a RAM 132 that "typically contains data and/or program modules that are immediately accessible to and/or presently being operated on by processing unit 120;" and "operating system 134, application programs 135, other program modules 136, and program data 137." APPLE1007, ¶[0036]; APPLE1003, ¶[139].

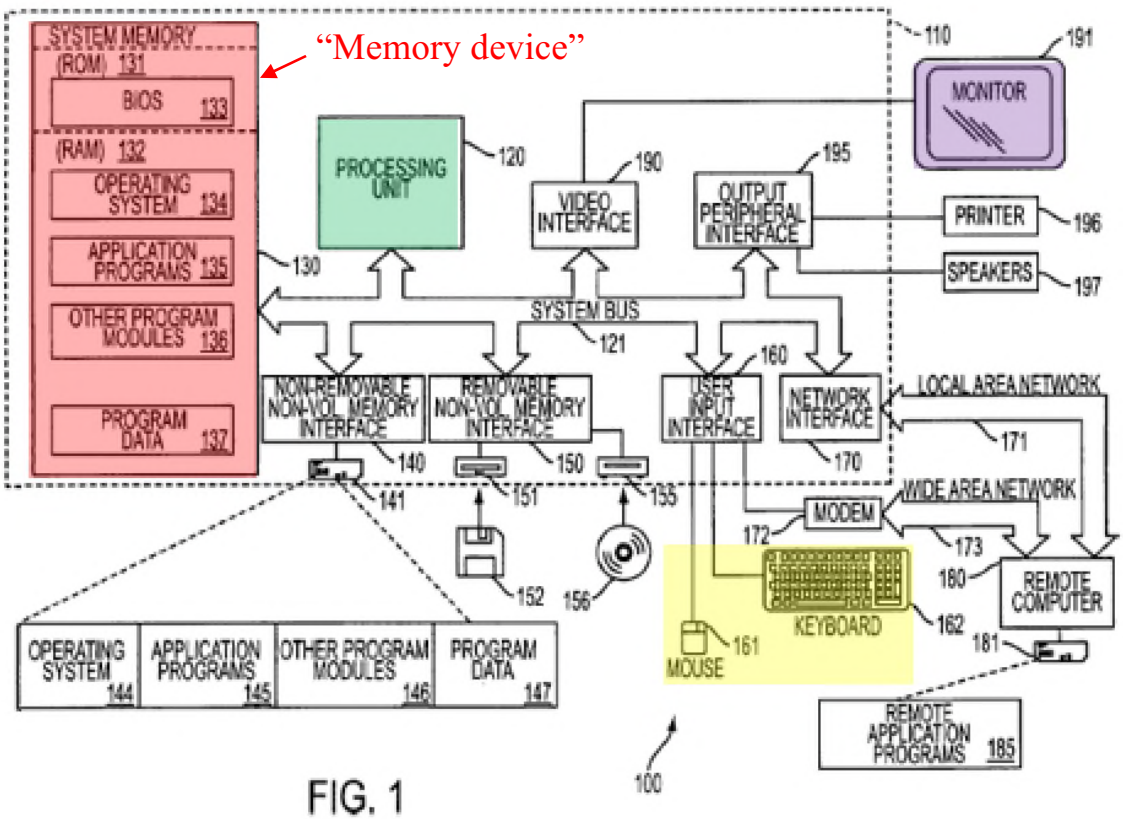


FIG. 1

APPLE1007, FIG. 1.

The memory device can include computer-readable medium for storage of information such as computer readable instructions, data structures, and program modules. APPLE1007, ¶[0035]. "A basic input/output system 133 (BIOS), containing the basic routines that help to transfer information between elements within computer 110, … is typically stored in ROM 131." APPLE1007, ¶[0036]. It would have been obvious to a POSITA that for HAC's system to operate, Anthony's processor would have executed operations according to instructions or code that are stored in a memory device, such as system memory 130, which provides instructions for the processor and applications to execute operations, e.g., operations [10e-1]-[10i-4]. APPLE1003, ¶[140].

[10d-1]

See supra [1a]; APPLE1003, ¶[141].

[10d-2]

See supra [1b]; APPLE1003, ¶[142].

[10d-3]

See supra [1c]; APPLE1003, ¶[143].

[10d-3i]

See supra [1c-1]; APPLE1003, ¶[144].

[10d-3ii]

See supra [1c-2]; APPLE1003, ¶[145].

[10d-3iii]

See supra [1c-3]; APPLE1003, ¶[146].

[10d-3iv]

See supra [1c-4]; APPLE1003, ¶[147].

[10d-3v]

See supra [1c-5]; APPLE1003, ¶[148].

[10d-4]

See supra [1d]; APPLE1003, ¶[149].

[10d-4i]

See supra [1d-1]; APPLE1003, ¶[150].

[10d-4ii]

See supra [1d-2]; APPLE1003, ¶[151].

[10d-4iii]

See supra [1d-3]; APPLE1003, ¶[152].

[10d-4iv]

See supra [1d-4]; APPLE1003, ¶[153].

Claim 11 - ***[11]***

See supra [3]; APPLE1003, ¶[154].

Claim 12 - ***[12]***

Hanggie discloses that its windows may be periodically updated. For

example. Hanggie teaches that "a change to the window icon or caption text may also necessitate a redraw of the CDWM-managed icon or caption content object, respectively, associated with the window." APPLE1006, ¶[0055]. When a video is being rendered in a video, Hanggie teaches that "the UCE processes incoming composition and resource updates from the CDWM, and **at intervals synchronized with the video refresh rates of each active video graphics adapter participating in the composition of the desktop, re-renders the desktop** (or the appropriate portion thereof in a multiple-display configuration) to a display-sized backing buffer. This is accomplished using the immediate-mode rendering services provided by a 3D graphics engine (such as Microsoft Direct3D®, which in turn transfers the desktop to a primary display surface." APPLE1006, ¶[0056]. As is known in the art, video refresh rates are periodic. APPLE1003, ¶[155]. Accordingly, it would have been obvious to a POSITA that Hanggie's disclosure of re-rendering the desktop at intervals synchronized with the video refresh rates would have included "periodically replacing at least one of said plurality of images with at least one other image, said other image showing at least one change to application specific data of a corresponding to one of said plurality of applications while said plurality of images are being displayed in said 3D space." APPLE1003, ¶[155]. It would also have been obvious to a POSITA that a composition and resource update, such as a change in video composition data,

corresponds to "at least one change to application specific data" when one of the displayed windows is showing a video. APPLE1003, ¶[155].

Claim 13 - ***[13]***

See supra [1c-2]; APPLE1003, ¶[156].

Claim 14 - ***[14]***

See supra [5]; APPLE1003, ¶¶[157].

Claim 18 - ***[18]***

See supra [9]; APPLE1003, ¶[163].

Claim 19

[19pre]

See supra [1pre]; APPLE1003, ¶[164].

[19a]

See supra [1a]; APPLE1003, ¶[165].

[19b]

See supra [1b]; APPLE1003, ¶[166].

[19c]

See supra [1c]; APPLE1003, ¶[167].

[19c-1]

See supra [1c-1]; APPLE1003, ¶[168].

[19c-2]

See supra [10h-2], [1c-2]; APPLE1003, ¶[169].

[19c-3]

See supra [1c-3]; APPLE1003, ¶[170].

[19c-4]

See supra [1c-4]; APPLE1003, ¶[171].

[19c-5]

See supra [1c-5]; APPLE1003, ¶[172].

[19d]

See supra [1d]; APPLE1003, ¶[173].

[19d-1]

See supra [1d-1]; APPLE1003, ¶[174].

[19d-2]

See supra [1d-2]; APPLE1003, ¶[175].

[19d-3]

See supra [1d-3]; APPLE1003, ¶[176].

[19d-4]

See supra [1d-4]; APPLE1003, ¶[177].

Claim 20 - ***[20]***

See supra [9]; APPLE1003, ¶[178].

B. GROUNDS 1B/2B – Claims 6-8 and 15-17 is obvious over HAC and Matthews

1. Matthews

Matthews discloses transforming a work area using a 3D transformation to move a displayed work area, such as a desktop with open windows, to reveal additional information. APPLE1008, Abstract; *see also infra* [8a] (pgs. 85-89). In particular, Matthews discloses a 3D rendering system that renders an **image of a work area 302 that includes an image of a desktop 307 and images of one or more open windows 305** can be tilted away from the user in response to a user selection of a button, such as the Start button 306. APPLE1008, ¶¶[0049]-[0056], [0059]-[0060], FIGS. 3A, 3B, 4A, 5A. The work area 302, desktop 307, and window(s) 305 "are virtually depicted in 3D space" in presentation area 303. APPLE1008, ¶[0058]; APPLE1003, ¶[74].

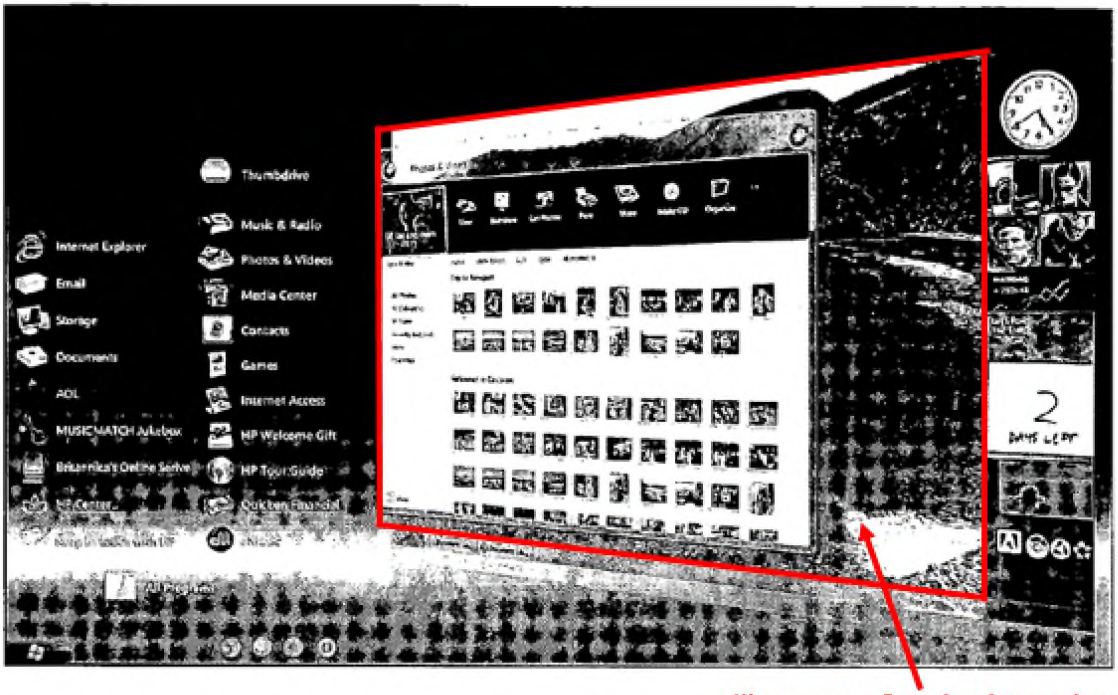


FIG. 3A

APPLE1008, FIG. 3A.

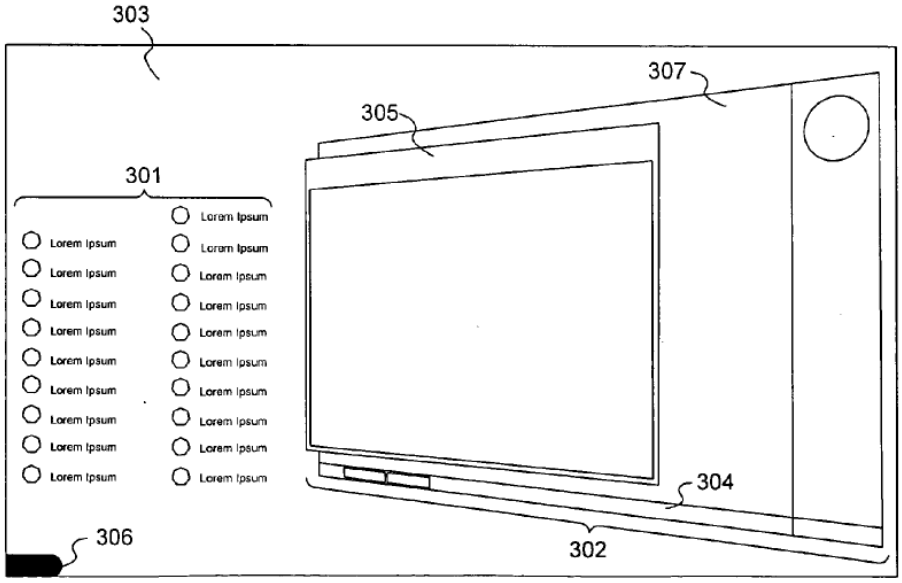
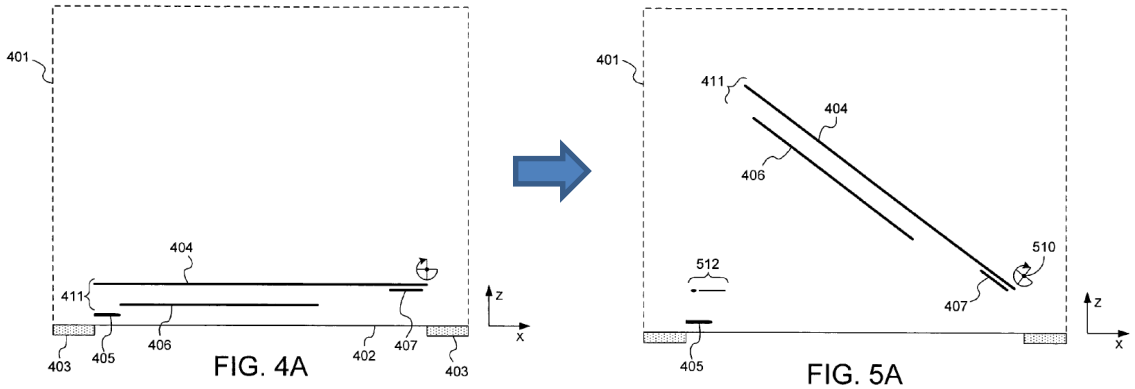


FIG. 3B

APPLE1008, FIG. 3B.



APPLE1008, FIG. 4A (left), FIG. 5A (right).

2. Combination of Matthews with HAC

Matthews explicitly teaches that Matthews' invention is to be used in Hanggie's CWDM system and in doing so, explicitly notes Hanggie's patent application number. APPLE1008, ¶[0041]. Matthews' FIGS. 1A and 1B, which describe the operating environment and system components, are the same as Hanggie's FIGS. 1A and 1B, and Hanggie's FIG. 1A is also almost identical to Anthony's FIG. 1. Thus, not only would it have been obvious and predictable to combine the teachings of the three references, but Matthews explicitly discloses that its teachings can be combined and implemented in Hanggie's system. APPLE1003, ¶[75].

A POSITA would have combined Matthews's teachings with the Anthony-Hanggie system so that information hidden behind a work area can be revealed by the tilting of a desktop display. APPLE1008, ¶¶[0010], [0011]; APPLE1003, ¶[76]. Moreover, Matthews' teachings "decouple[ ] the operating system from the applications … and prevents visual clutter while taking full advantage of the

graphics capabilities of the host computer." APPLE1008, ¶[0052].

A POSITA would have combined the teachings of HAC and Matthews because doing so would merely involve combining prior art elements according to known methods to yield predictable results. APPLE1003, ¶[77]. Here, as called for by Matthews, Matthews' known method would have been combined with Hanggie's CWDM system in HAC's system to yield the predictable result of rendering a 3D display with a transformable desktop display. APPLE1003, ¶[77].

3. Manner in which HAC and Matthews render claims 6-8 and 15-17 obvious

Claim 6

[6a]

Matthews discloses a method and apparatus for transforming a work area using a 3D transformation to move a displayed work area, such as a desktop with open windows, to reveal additional information. APPLE1008, Abstract. In particular, Matthews discloses that an **image of a work area 302 that includes an image of a desktop 307 and images of one or more open windows 305** can be tilted away from the user in response to a user selection of a button, such as the Start button 306. APPLE1008, ¶¶[0049]-[0050], [0056], [0060], FIGS. 3A, 3B, 4A, 5A (reproduced below). The work area 302, desktop 307, and window(s) 305 "are virtually depicted in 3D space" in presentation area 303. APPLE1008, ¶[0058]; APPLE1003, ¶[125].

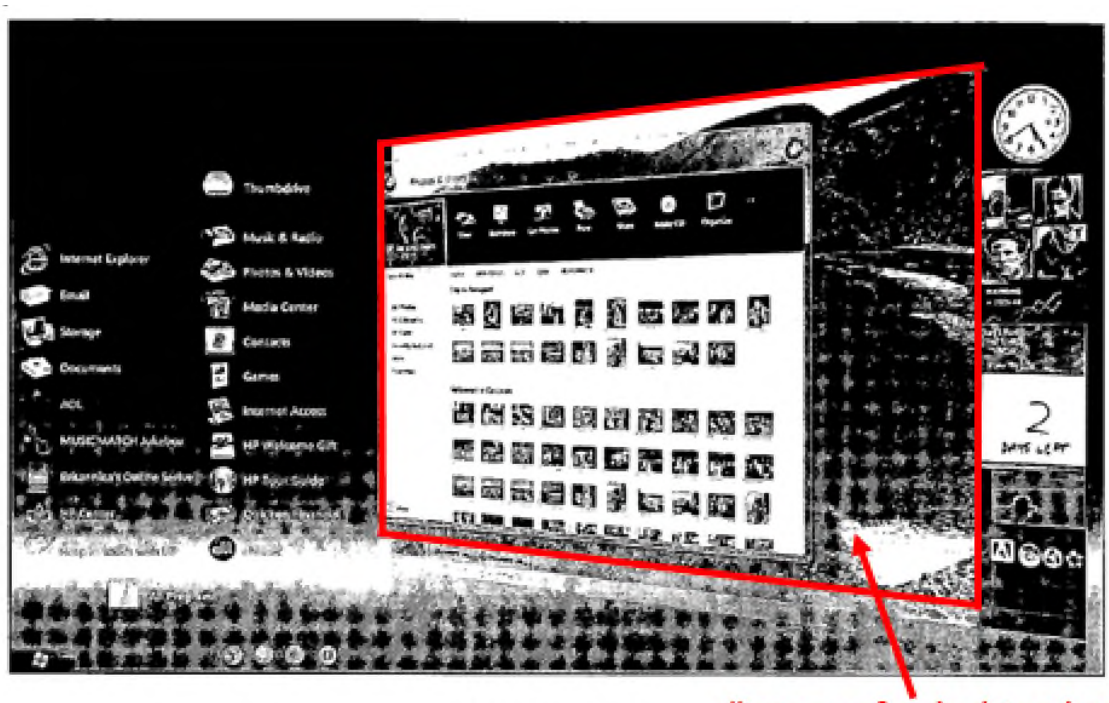


FIG. 3A

APPLE1008, FIG. 3A.

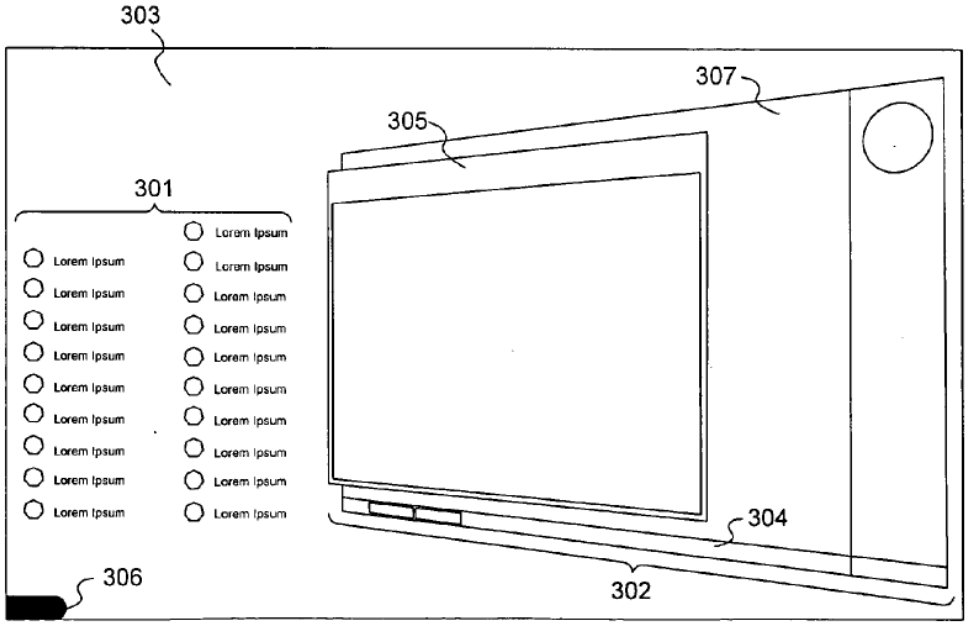
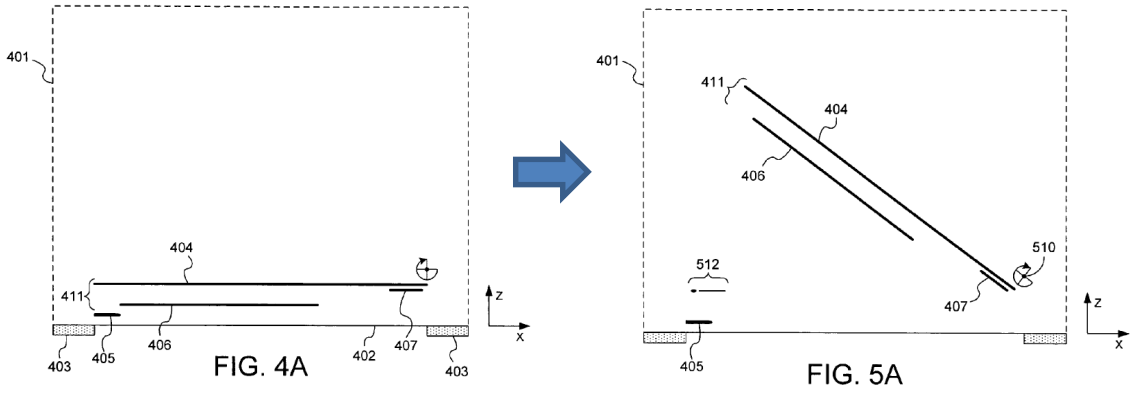


FIG. 3B

APPLE1008, FIG. 3B.



APPLE1008, FIG. 4A (untransformed display) (left), FIG. 5A (transformed display) (right).

[6b]

Once in the transformed display (*see* FIGS. 5A and 3B above), Matthews teaches that a user can click on the work area 302 to return to an untransformed display to focus on applications in the work area. APPLE1008, ¶[0051]. A user can also select a particular application window and continue working on that application window after the work area is untransformed. APPLE1008, ¶[0053] ("At a minimum, if the user points the mouse in the work area and clicks, the work area can be returned to its initial un-transformed state, and the user may resume normal manipulation of the work area."), ¶[0070]; APPLE1003, ¶[126].

As can be appreciated from the foregoing, Matthews teaches that by clicking on the image of the desktop in the transformed work area 302 and untransforming the work area 302, the computer display would then "replac[e] said fourth image with said desktop in response to said selection" of the user. APPLE1003, ¶[127].

Matthews explains that the user can then "resume normal manipulation of the work area." APPLE1008, ¶[0053]. It would have been obvious to a POSITA that the "normal manipulation of the work area" includes "receiving an interaction from said user with said other application, and opening said other application in response to said interaction" when a user selects an icon representing an application to open the application window, as described above in [1b] and [1c]-[1c-5]. APPLE1003, ¶[127]. For example, Anthony discloses that "[i]f an icon 606a in the focal group 509 represents a file, double-clicking the file will cause the system to attempt to open the file." APPLE1007, ¶[0047]. In this manner, the HAC-Matthews combination renders obvious "allowing said user to open said other application," as recited in [6b]; APPLE1003, ¶[127].

*Claim 7 - **[7]***

Matthews gives examples of a web browser application, such as the Internet Explorer, as one example of a window that can be opened in a computer. APPLE1008, FIGS. 2, 8, 9, ¶[0070]. Although not explicitly stated, it would have been obvious to a POSITA that when a user desires to open three web pages in a new window, the HAC-Matthews system would have displayed three windows corresponding to three web browser applications such that "each one of said first, second, and third objects comprises a web browser window." APPLE1003, ¶[128].

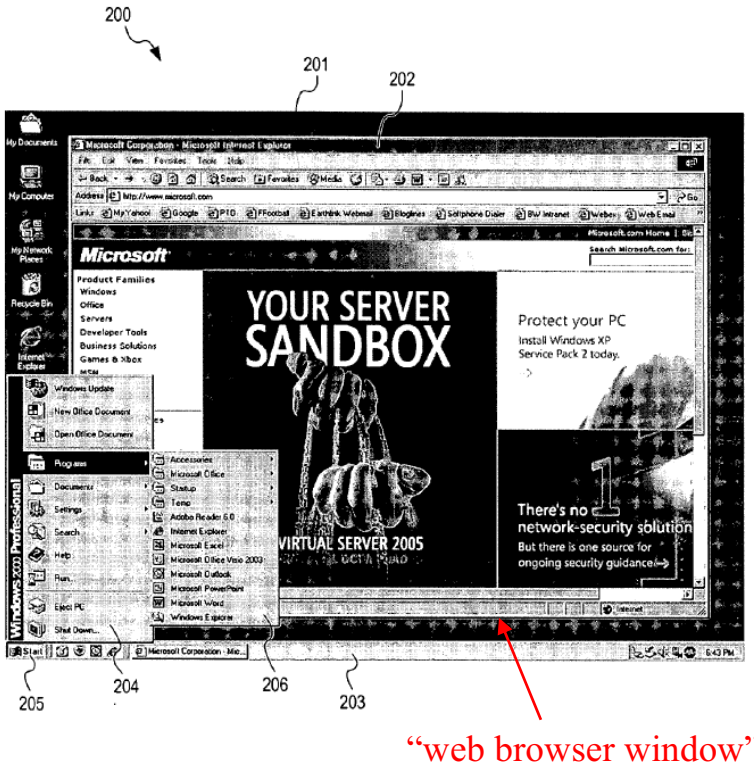


FIG. 2
(PRIOR ART)

APPLE1008, FIG. 2.

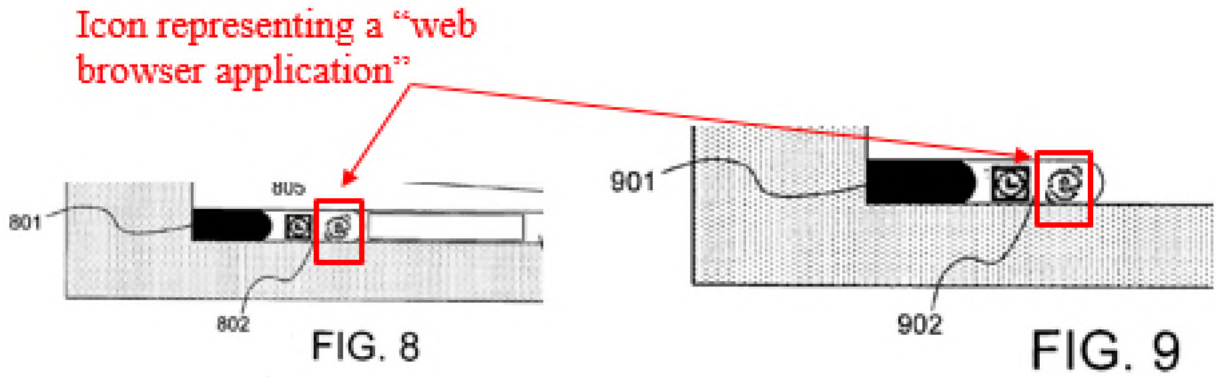


FIG. 8 FIG. 9

Portion of APPLE1008, FIGS. 8 (left), 9 (right).

Claim 8

[8a]

As explained in [6a] and [6b], Matthews discloses transforming a computer display to a second 3D immersive space (presentation area 303) that includes an image of a work area 302 that includes images of a desktop 307 and one or more windows 305. APPLE1008, ¶¶[0049]-[0050], [0056], [0060], FIGS. 3A, 3B, 4A, 5A (reproduced below); APPLE1003, ¶[129].

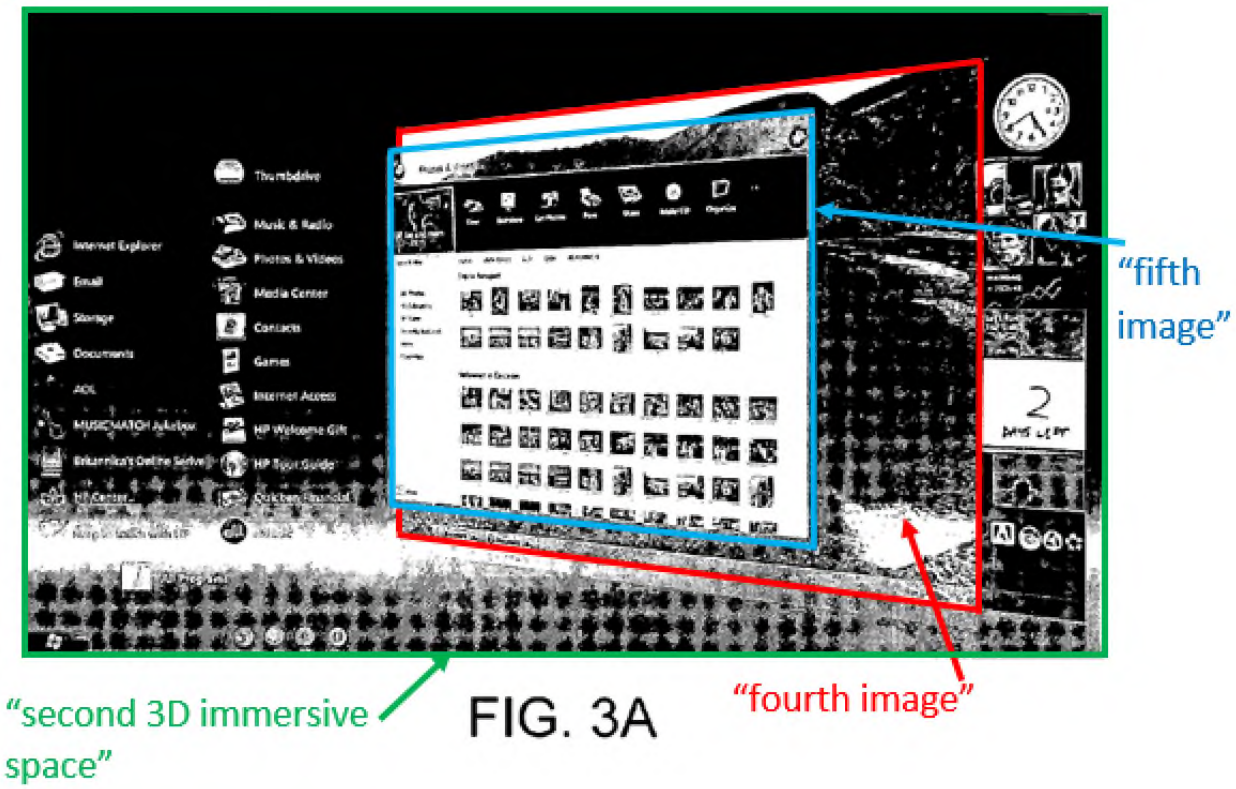


FIG. 3A

APPLE1008, FIG. 3A.

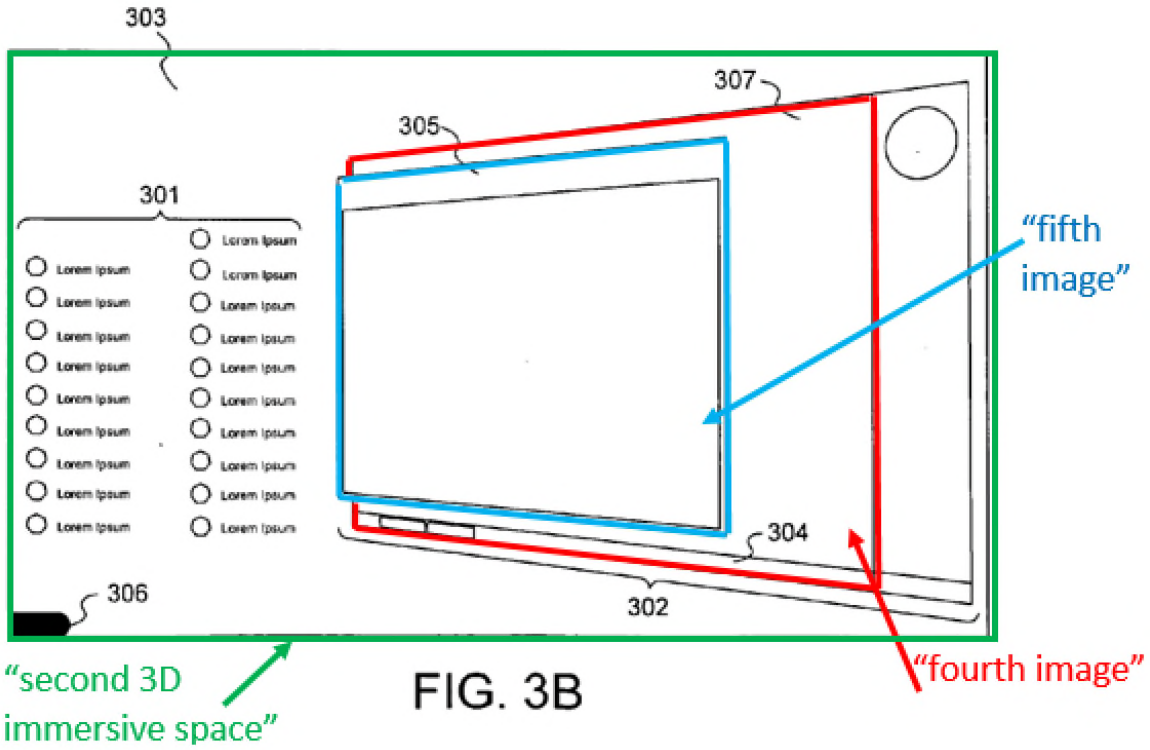


FIG. 3B

APPLE1008, FIG. 3B.

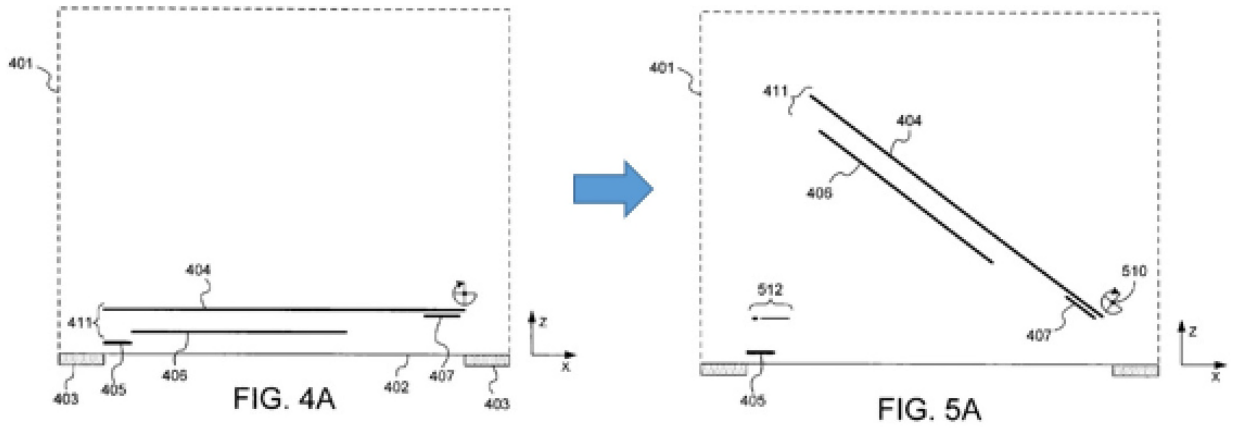


FIG. 4A FIG. 5A

APPLE1008, FIG. 4A (left), FIG. 5A (right).

FIG. 7A more clearly illustrates the mapping of the fourth and fifth images. APPLE1003, ¶[130]. As shown in FIG. 7A, the fourth image includes a plurality of images, such as images 704, 706a, and 706b. The transformed space

corresponds to the "second 3D immersive space." When multiple windows are displayed, e.g., 4 application windows, it would have been obvious to a POSITA that the four application windows would form part of the fourth image similar to the manner in which windows 706a and 706b are displayed. APPLE1003, ¶[130]. Moreover, it would also have been obvious to a POSITA that when a fourth application (different from the other three open applications), such as a picture viewer, is previously opened by a user, a fifth image, such as image 706a, is displayed and is also a portion of the fourth object and is generated by a fourth application (e.g., picture viewer) previously opened by said user. APPLE1003, ¶[130].

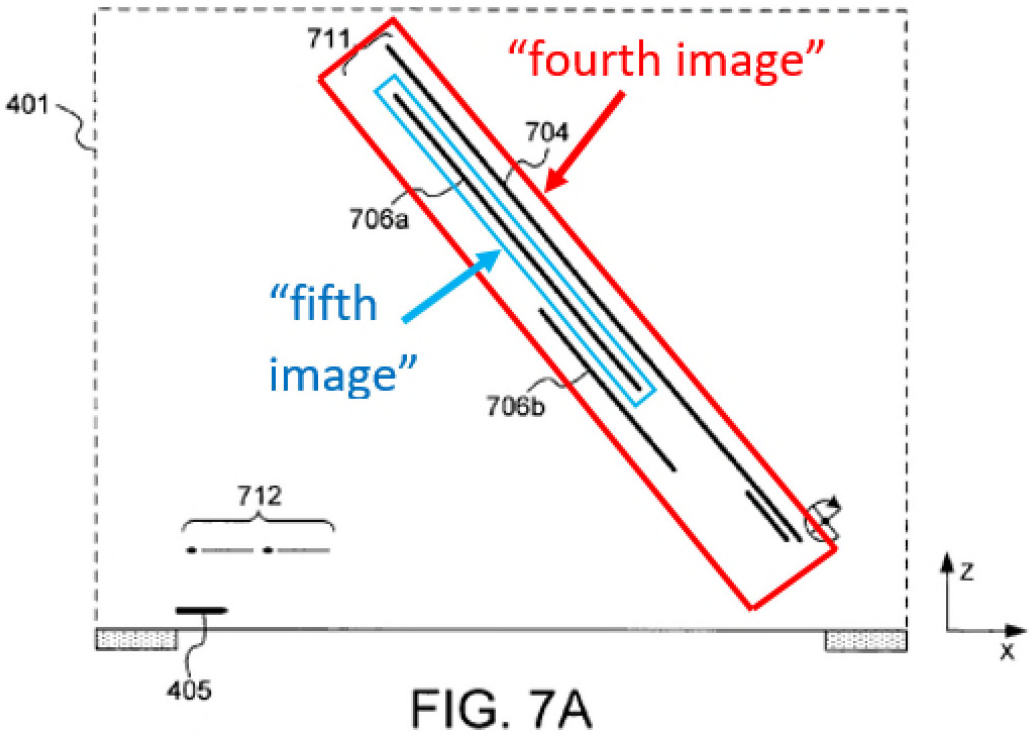
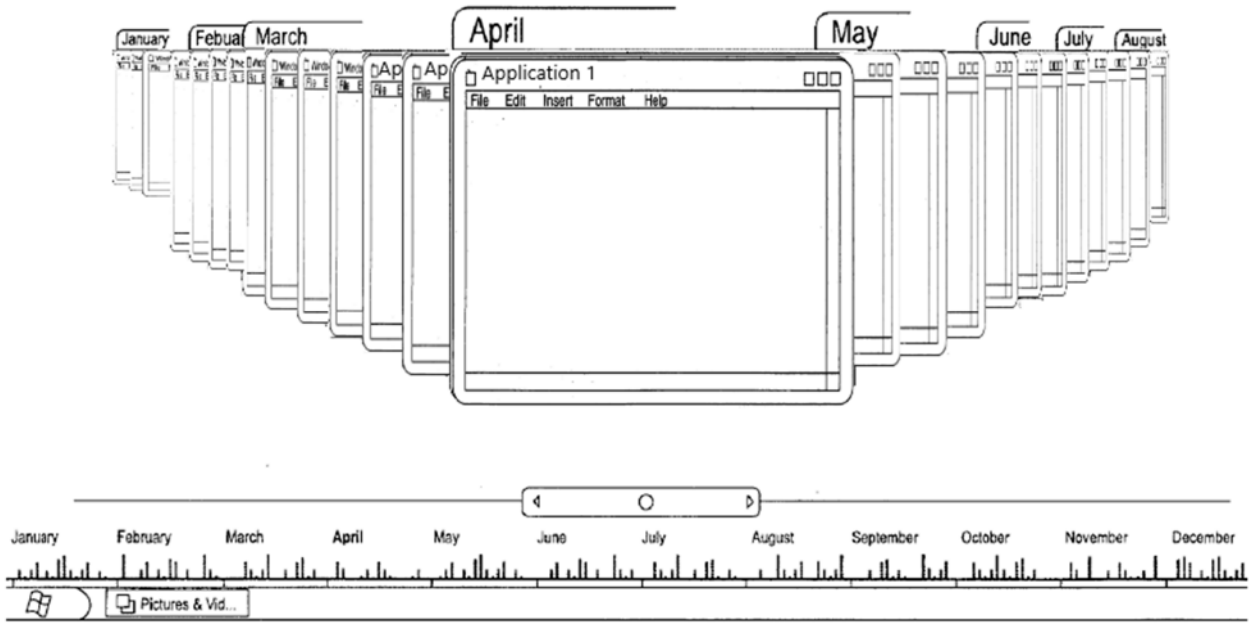


FIG. 7A

APPLE1008, FIG. 7A.

In the HAC-Matthews combination, it would have been obvious to a POSITA that multiple windows would be displayed in a similar manner as windows 706a, 706b but in a timeline, as shown in the combined HAC figure. APPLE1003, ¶[131]. The timeline would have been tilted in the manner shown in Matthew's FIG. 7A when in the transformed display mode. APPLE1003, ¶[131].



APPLE1003, pg. 96.

Matthews explains that the presentation area 303 is a second 3D immersive space by explicitly stating that "a three dimensional rendering system is used here" and that the "[t]he presentation area [ ] is virtual …; it is a virtual 3D space in which items can be presented to a user." APPLE1008, ¶¶[0052], [0054], [0059]. Matthews also teaches that "[t]ransforming the presently displayed work area can be accomplished using a 3D graphics system." APPLE1008, ¶[0052]. This system

is described as using a 3D graphics video card, a series of 3D meshes, a high level graphics API, such as Java 3D™, and X, Y, and Z coordinates to render a 3D scene. APPLE1008, ¶[0055]; APPLE1003, ¶[132].

[8b]

See supra [6b], [1b], [1c]; APPLE1003, ¶[133].

Claim 15

[15a]

See supra [6a]; APPLE1003, ¶[158].

[15b]

See supra [6b]; APPLE1003, ¶[159].

Claim 16 - ***[16]***

See supra [7]; APPLE1003, ¶[160].

Claim 17

[17a]

See supra [8a]; APPLE1003, ¶[161].

[17b]

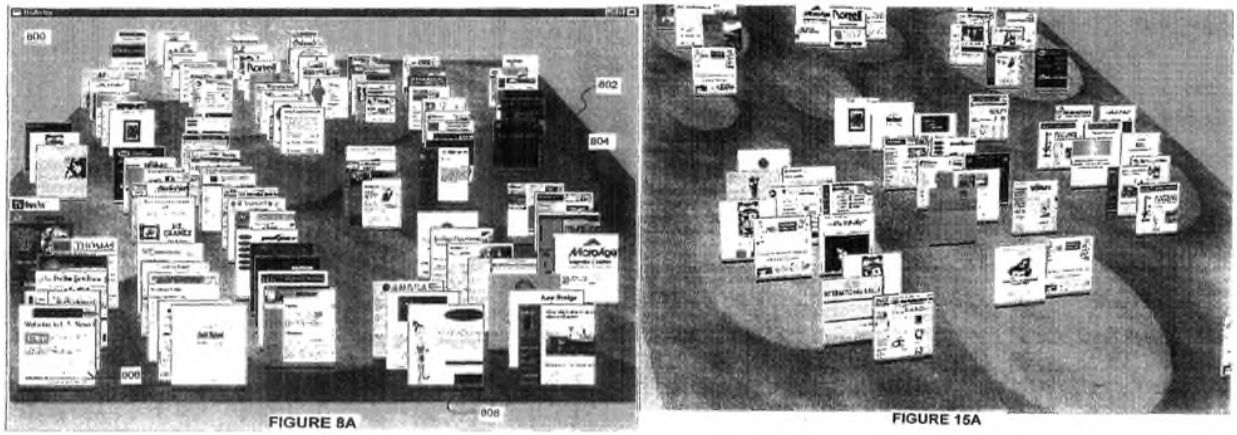
See supra [8b]; APPLE1003, ¶[162].

C. GROUNDS 1C/2C – Claim 7 is obvious over HAC, Matthews, and Robertson

1. Robertson

Robertson discloses an interface that "exploit[s] spatial memory" by "simulat[ing] three dimensions" and representing webpages in the 3D space as object thumbnails bearing a low resolution image of the corresponding content. APPLE1009, 6:15-28; *see also id*., 6:30-67, 9:11-50, 12:54-13:4 ("the object thumbnails 806 represent web (or hypertext markup language or 'HTML') pages"); APPLE1003, ¶¶[78]-[79].

Robertson's Figures 8A and 15A (below) depict two exemplary 3D-GUIs in which webpages are represented by object thumbnails on a simulated 3D landscape. APPLE1009, 12:54-13:4, 17:21-45, Figures 8A-18; APPLE1003, ¶[80].



APPLE1009, FIG. 8A (left), FIG. 15A (right).

Robertson provides "'live' objects within an associated application" "for editing or otherwise working on a selected object." APPLE1009, 13:55-67. For

example, when a thumbnail is selected by a user, "the Internet Explorer browser…render[s] [the corresponding] web page, with the [3D] user interface…in the background." APPLE1009, 13:67-14:14. "The application can be maximized…to substantially fill the screen of the video monitor." *Id*.; APPLE1003, ¶[81].

2. Combination of Robertson with HAC-Matthews

As disclosed by Robertson, using an Internet Explorer window to browse the Internet and access information was well known in the art. APPLE1009, 2:26-67; APPLE1003, ¶[82]. To avoid "navigating through a hierarchy of menus, or entering a search query," it would have been obvious to a POSITA to display multiple browser application windows (e.g., 4 browser windows when visiting 4 web pages) in a 3D display. APPLE1009, 3:36-46; APPLE1003, ¶[82]. Displaying multiple browsers is also beneficial because the user can avoid wasting time trying to find the relevant bookmark from among multiple bookmarks. APPLE1009, 3:47-66. For at least these reasons, the Anthony, Hanggie, Matthews, and Robertson system renders obvious [7]. APPLE1003, ¶[82].

Anthony, Hanggie, and Matthews render obvious multiple windows being displayed in 3D environment. Robertson explains that, for a user interesting in browsing the Internet, it would have been obvious to display the multiple windows are separate web browsers corresponding to web browser applications used to

browse the Internet. APPLE1009, FIGS. 8A, FIG. 15A; APPLE1003, ¶[83].

Combining the teachings of Anthony, Hanggie, Matthews, and Robertson would have been obvious as it merely involves combining prior art elements according to known methods to yield predictable results. APPLE1003, ¶[84]. Here, the HAC-Matthews system would have been used to display web browser windows, as disclosed by Robertson, to yield the predictable result that a user can view multiple browser windows at one time. APPLE1003, ¶[84]. The windows would have been arranged in a chronological order, as taught by Anthony, so that the user can see which windows the user interacted with at what time and can then also more easily navigate to the window of interest. APPLE1003, ¶[84].

Moreover, all of the references are directed to rendering windows in 3D space and have the same assignee (Microsoft). Indeed, Robertson's computer system (shown in FIG. 1A) is similar to Anthony, Hanggie, and Matthews' FIG. 1A, thus making the combination of these references predictable and foreseeable. APPLE1003, ¶[85].

3. Manner in which HAC, Matthews, and Robertson render obvious claim 7

*Claim 7 - **[7]***

See supra §VI.C.1-§VI.C.2.

VII. PTAB DISCRETION SHOULD NOT PRECLUDE INSTITUTION

A. 35 U.S.C. §325(d)—*Advanced Bionics*

Advanced Bionics and the *Becton* factors strongly favor institution. [*Advanced Bionics LLC v. MED-EL Elektromedizinische Gerate GmbH*, IPR2019-01469, Paper 6 (PTAB Feb. 13, 2020) ("*Advanced Bionics*") (precedential); *Becton, Dickinson and Co. v. B. Braun Melsungen AG*, IPR2017-01586, Paper 8 (PTAB Dec. 15 2017) ("*Becton*") (precedential).

The examiner never issued a rejection during the '868 Patent's prosecution. *See*, *generally*, APPLE1002. And there is no indication in the '868 Patent's file history that the examiner substantively considered any of the prior art applied in this Petition prior to allowing the application that issued as the '868 Patent. *Id*. Because the '868 Patent's prosecution did not involve negotiation between the applicant and the examiner regarding the teachings of the prior art applied in the instant Petition, the instant Petition's art and arguments cannot be said to be the same as, or substantially similar to, art and arguments previously presented to the Office in connection with the '868 Patent. *Id*. Accordingly, neither condition of the first prong of the *Advanced Bionics* framework is met, and there is no need to reach the second prong to resolve against discretionary denial under Section 325(d). *See*, *e.g.*, *Oticon Medical AB et. al. v. Cochlear Ltd.*, IPR2019-00975, Paper 15 at 20 (PTAB Oct. 16, 2019) (precedential) ("There is new, noncumulative prior art asserted in the

Petition...[f]or at least this reason, we determine not to exercise our discretion under § 325(d)").

Moreover, the Challenged Claims were allowed based on the Examiner's mistaken view that the prior art failed to teach features [1c]-[1c-2]. APPLE1002, 27. But these features were plainly disclosed within the prior art, including within the art applied within this Petition in §VI. Thus, the Examiner's failure to consider obviousness of the Challenged Claims over uncited references such as Hanggie, Anthony, Matthews, and Robertson constitutes material error.

For at least this additional reason, Petitioner respectfully submits that discretionary denial is unwarranted, and that the Board should instead institute IPR.

B. 35 U.S.C. §314(a)—*Fintiv*

Given that Google is not a Defendant in the parallel proceeding, this favors institution. As to LG, the Congressional intent, the goals of Fintiv, and the interim guidance issued by Director Vidal ("Director's Guidance" or APPLE1021) all favor institution. *Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 11, 6 (PTAB Mar. 20, 2020) (precedential) ("*Fintiv*"). Petitioner asks the Board to consider the challenges raised in this Petition, which mirror the challenges raised in Apple's prior petition.

First and foremost, if Apple's petition has already been instituted, a *Fintiv* denial would be inappropriate because (1) a reasonable likelihood of success has

already been found for the presented grounds and (2) joinder petitioners that assume an understudy role consume only minimal resources of the Board and Patent Owner.

If Apple's petition terminates prior to institution, the merits of the Petition are still "compelling," which "alone demonstrates that the PTAB should not discretionarily deny institution under *Fintiv*." [*Director's Guidance*, 3-5]. As further explained below, even if the Board addresses the full *Fintiv* framework, a holistic analysis favors institution.

1. Factor: Petitioner Will Request A Stay

Institution will enable the Board to resolve the issue of validity, and a finding of invalidity will relieve the Eastern District of Texas of the need to continue with the parallel litigation. If the Apple IPR, or this IPR, is instituted, LG Electronics, Inc. and LG Electronics U.S.A., Inc. (collectively the "LG petitioners") will move to stay the District Court case. Factor 1 favors institution.

2. Factor 2: District Court Trial Dates Are Unreliable

While the court set trial for October 16, 2023, LGE1028 the Director recognizes "that scheduled trial dates are unreliable and often change." [*Director's Guidance*, 8]. Thus, "the proximity to trial should not along outweigh" other relevant factors. [*Id.*, 8]. Moreover, the LG petitioners intend to request a stay if either this or the Apple IPR is instituted. Factor 2 is neutral.

3. Factor 3: Parallel Proceeding Considerations

In the District Court, a *Markman* order has not issued, expert reports have not been exchanged, and fact discovery is ongoing. The Google petitioners are not a party in the District Court case, which strongly weighs against discretionary denial. Factor 3 favors institution.

4. Factor 4: The Petition's Grounds Are Materially Different From Any That Might Be Raised in the Litigation

To eliminate any doubt as to the absence of meaningful overlap between the litigation and IPR, the LG petitioners stipulate that, unless the Board denies or later vacates institution of this Petition, the LG petitioners will not seek resolution in the District Court of invalidity based on the specific grounds asserted in this Petition. *Sand Revolution*, 11-12. Factor 4 favors institution.

5. Factor 5: Parties in Parallel Proceedings

The Petition is potentially helpful to future defendants. Institution would serve overall efficiency, enabling the Board to determine invalidity of claims that Patent Owner might otherwise later assert against others. Factor 5 favors institution.

6. Factor 6: Other Considerations

Director Vidal has explained that "the PTAB will not deny institution...under *Fintiv*...when a petition presents compelling evidence of unpatentability...." [*Director's Guidance*, 3-5, 9; *see also Fintiv*, 14-15].

The merits of this Petition are compelling. As demonstrated in the Petition, institution would result in a finding of unpatentability of the Challenged Claims, which are obvious based on prior art references that materially differ from those considered by the Examiner during prosecution.

Even assuming that the litigation is thought relevant (it is not), the strength of the merits alone is enough to outweigh any inefficiencies born of parallel litigation. [*Director's Guidance*, 3-5, 9; *Fintiv*, 14-15].

Moreover, institution would *promote* the AIA's objectives of providing an effective and efficient alternative to district court litigation. *See* Sen. Rep. No. 110-259 (2008); H.R. Rep. No. 112-98, pt. 1, pp. 39-40.

Accordingly, *Fintiv* Factor 6 strongly favors institution. *See Snap*, 16-19.

C. 35 U.S.C. §314(a) – General Plastic

Petitioner respectfully submits application of the *General Plastic* analysis is inapplicable here. In *General Plastic*, the Board set forth a series of factors that may be analyzed for follow-on petitions to help conserve the finite resources of the Board. Here, both Petitioner and Apple submitted separate, independent petitions.

In the current motion, Petitioner merely seeks to join Apple's petition and does not present any new grounds. Petitioner has not previously submitted an IPR on the '868 patent. Petitioner would be taking an understudy role and the Board's finite resources would not be impacted. Moreover, there is no strategic advantage to be gained by filing this additional petition, and there are no concerns of "road mapping" the Patent Owner's strategy because Petitioner has submitted a petition that is substantively identical to Apple's petition.

VIII. CONCLUSION

Petitioner respectfully requests institution.

IX. PAYMENT OF FEES – 37 C.F.R. §42.103

The undersigned authorizes the Director to charge the fee specified by 37 C.F.R. § 42.15(a) and any additional fees that might be due in connection with this Petition to Deposit Account No. 50-3266.

X. MANDATORY NOTICES – 37 C.F.R. §42.8(a)(1)

A. Real Party-in-Interest – 37 C.F.R. §42.8(b)(1)

LG Electronics Inc., LG Electronics U.S.A., Inc., and Google LLC are the real parties-in-interest.

B. Related Matters – 37 C.F.R. §42.8(b)(2)

The '868 patent is the subject of the following proceedings:

Caption	Case No.	Court	Filing Date	Status
SpaceTime3D, Inc. v. LG Electronics, Inc, et al.	2-22-cv-00049	EDTX	2022-02-10	Active
SpaceTime3D, Inc. v. Apple Inc.	6:22-cv-149	WDTX	2022-02-10	Active
Apple Inc. v. SpaceTime3D, Inc.	IPR2023-00344	PTAB	2022-12-13	Active
SpaceTime3D, Inc. v. Samsung Electronics Co., Ltd. et al.	2:19-cv-00372	EDTX	2019-11-14	Terminated
Samsung Electronics Co., Ltd. v. SpaceTime3D, Inc.	IPR2020-01419	PTAB	2020-08-04	Terminated Pre-Institution

C. Lead and Back-Up Counsel – 37 C.F.R. §42.8(b)(3)

Lead counsel:

Matthew Satchwell (Reg. No. 58,870)
DLA Piper LLP (US)
444 W Lake St. #900
Chicago, IL 60606
matthew.satchwell@dlapiper.com
312-368-2111 (tel.)
312-731-4403 (fax)

Backup counsel:

Brent Yamashita (Reg. No. 53,808)
DLA Piper LLP (US)
2000 University Ave.
East Palo Alto, CA 94303
brent.yamashita@dlapiper.com
650-833-2348 (tel.)
650-687-1206 (fax)

Christopher Duerden (Reg. No. 77,464)

DLA Piper LLP (US)
33 Arch St #26
Boston, MA 02110
chris.duerden@dlapiper.com
617-406-6077 (tel.).

Erika H. Arner (Reg. No. 57,540)
erika.arner@finnegan.com
Finnegan, Henderson, Farabow, Garrett, & Dunner LLP
1875 Explorer Street, Suite 800
Reston, VA 20190-6023
Tel: 571-203-2700
Fax: 202-408-4400

Cory C. Bell (Reg. No. 75,096)
cory.bell@finnegan.com
Finnegan, Henderson, Farabow,
Garrett & Dunner, LLP
2 Seaport Lane, 6th Floor
Boston, MA 02210-2001
Tel: 617-646-1600
Fax: 202-408-4400

Yanyi Liu (Reg. No. 71,046)
yanyi.liu@finnegan.com
Finnegan, Henderson, Farabow,
Garrett & Dunner, LLP
Stanford Research Park
3300 Hillview Avenue
Palo Alto, CA 94304-1203
Tel: 650-849-6600
Fax: 202-408-4400

Yinan Liu (Reg. No. 72,482)
yinan.liu@finnegan.com
Finnegan, Henderson, Farabow,
Garrett & Dunner, LLP
2 Seaport Lane, 6th Floor
Boston, MA 02210-2001
Tel: 617-646-1600

Fax: 202-408-4400

D. Service Information

Service information for lead and backup counsel is provided in the designation of lead and backup counsel above.

Dated: February 10, 2023

Respectfully submitted,

/Matthew Satchwell/

Reg. No. 58,870
DLA Piper LLP (US)
444 W Lake St. #900
Chicago, IL 60606
matthew.satchwell@dlapiper.com
Phone: 312-368-2111
Fax: 312-731-4403

CERTIFICATION UNDER 37 CFR § 42.24(d)

Under the provisions of 37 CFR § 42.24(d), the undersigned hereby certifies that the applicable word count for the foregoing Petition for *Inter Partes* Review totals 13,990 words, which is less than the 14,000 allowed under 37 CFR § 42.24(a)(i).

Dated: February 10, 2023

Respectfully submitted,

/Matthew Satchwell/

Reg. No. 58,870
DLA Piper LLP (US)
444 W Lake St. #900
Chicago, IL 60606
matthew.satchwell@dlapiper.com
Phone: 312-368-2111
Fax: 312-731-4403

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. §§ 42.6(e) and 42.105(b), the undersigned hereby certifies that a copy of the foregoing petition and exhibits were provided to Patent Owner electronically, by agreement (*see* LGE1029), by emailing a copy of the same to the following attorneys for Patent Owner:

Meng Xi (mxi@susmangodfrey.com)
Max L. Tribble (mtribble@susmangodfrey.com)
Matthew R. Berry (mberry@susmangodfrey.com)
Gloria Park (gpark@susmangodfrey.com)
Floyd G. Short (fshort@susmangodfrey.com)
Adam Tisdall (atisdall@susmangodfrey.com)

In addition, a true and correct copy of the foregoing petition and exhibits were served via Priority Mail Express, on the following attorney of record for the patent at the following address:

Todd Fitzsimmons
Fitzsimmons IP Law
PO Box 199
Gardena, CA 90248

In addition, a copy of the foregoing petition and exhibits were electronically served in its entirety on counsel for Petitioner in related Case No. IPR2023-00242:

W. Karl Renner (renner@fr.com)
Andrew B. Patrick (patrick@fr.com)
Kenneth Wayne Darby, Jr. (kdarby@fr.com)
Usman A. Khan (khan@fr.com)

Dated: February 10, 2023

Respectfully submitted,

By:/ Matthew D. Satchwell /

Matthew D. Satchwell (Reg. No. 58,870)
Email: matthew.satchwell@dlapiper.com
DLA Piper LLP (US)
444 West Lake Street, Suite 900
Chicago, Illinois 60606
Telephone: (312) 368-2111
Fax: (312) 236-7516