PTO/AIA/01 (06-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

DECLARATION (37 CFR 1.63) FOR UTILITY OR DESIGN APPLICATION USING AN APPLICATION DATA SHEET (37 CFR 1.76)

Title of Invention	Methods, Systems, and Program Products for Sharing a Data Object in a Data Store Via a Communication

As the below named inventor, I hereby declare that:

This declaration is directed to:

- [x] The attached application, or
- [ ] United States application or PCT international application number _____

filed on _____.

The above-identified application was made or authorized to be made by me.

I believe that I am the original inventor or an original joint inventor of a claimed invention in the application.

I hereby acknowledge that any willful false statement made in this declaration is punishable under 18 U.S.C. 1001 by fine or imprisonment of not more than five (5) years, or both.

WARNING:

Petitioner/applicant is cautioned to avoid submitting personal information in documents filed in a patent application that may contribute to identity theft. Personal information such as social security numbers, bank account numbers, or credit card numbers (other than a check or credit card authorization form PTO-2038 submitted for payment purposes) is never required by the USPTO to support a petition or an application. If this type of personal information is included in documents submitted to the USPTO, petitioners/applicants should consider redacting such personal information from the documents before submitting them to the USPTO. Petitioner/applicant is advised that the record of a patent application is available to the public after publication of the application (unless a non-publication request in compliance with 37 CFR 1.213(a) is made in the application) or issuance of a patent. Furthermore, the record from an abandoned application may also be available to the public if the application is referenced in a published application or an issued patent (see 37 CFR 1.14). Checks and credit card authorization forms PTO-2038 submitted for payment purposes are not retained in the application file and therefore are not publicly available.

LEGAL NAME OF INVENTOR

Inventor: Robert Paul Morris Date (Optional): 2012-09-25

Signature: /Robert Paul Morris/

Note: An application data sheet (PTO/AIA/14 or equivalent), including naming the entire inventive entity, must accompany this form. Use an additional PTO/SB/AIA01 form for each additional inventor.

This collection of information is required by 35 U.S.C. 115 and 37 CFR 1.63. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 1 minute to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Privacy Act Statement

The **Privacy Act of 1974 (P.L. 93-579)** requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (*i.e.*, GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

METHODS, SYSTEMS, AND COMPUTER PROGRAM PRODUCTS FOR SHARING A DATA OBJECT IN A DATA STORE VIA A COMMUNICATION

RELATED APPLICATIONS

[0001] This application is related to the following commonly owned U.S. Patent Applications: Application No. 12/833,016 (Docket No 0158) filed on 2010/07/09, entitled "Methods, Systems, and Program Products for Referencing an Attachment in a Communication";

[0002] Application No. 12/833,014 (Docket No 0155) filed on 2010/07/09, entitled "Methods, Systems, and Program Products for Processing a Request for a Resource in a Communication";

[0003] Application No. 13/624,906 (Docket No 0149) filed on 2012/09/22, entitled "Methods, Systems, and Program Products for Processing a Data Object Identification Request in a Communication"; and

[0004] Application No. 13/624,940 (Docket No 0265) filed on 2012/09/23, entitled "Methods, Systems, and Program Products for Processing a Reference in a Communication to a Remote Data Object".

BACKGROUND

[0005] A variety of means exist for sharing filed, folders, and other data via network. For each a user must know how to configure and/or setup the sharing. For example, sharing a file a Samba file system is different than sharing a via cloud based storage system, such as DROPBOX. Sharing via email is still different. Sharing copies of files is perhaps the best know means and most widely used means for sharing copies. It would useful if sharing files, folders, directory data, and other network accessible data were as easy as sending an attachment in an email.

Motion Offense Exhibit 2004

[0006] Accordingly, there exists a need for methods, systems, and computer program products for sharing a data object in a data store via a communication.

SUMMARY

[0007] The following presents a simplified summary of the disclosure in order to provide a basic understanding to the reader. This summary is not an extensive overview of the disclosure and it does not identify key/critical elements of the invention or delineate the scope of the invention. Its sole purpose is to present some concepts disclosed herein in a simplified form as a prelude to the more detailed description that is presented later.

[0008] Methods and systems are described for sharing a data object in a data store via a communication. In one aspect, the method includes receiving, by a first communications agent in a first execution environment according to a first communication protocol via a network from a second communications agent in a second execution environment, a first message, in a communication and addressed to a first user represented by the first communications agent, that includes a mount descriptor that identifies a data object **in** a second data store in the second execution environment. The method further includes detecting the mount descriptor. The method still further includes determining a first location in a first data store in the first execution environment. The method additionally includes creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store. Performing at least one the preceding actions comprising the method includes execution of an instruction by a processor.

[0009] Also, a system for sharing a data object in a data store via a communication is described that includes one or more processors and logic encoded in one or more tangible media for execution by the one or more processors that when executed is operable for receiving, by a first communications agent in a first execution environment

Motion Offense Exhibit 2004

according to a first communication protocol via a network from a second communications agent in a second execution environment, a first message, in a communication and addressed to a first user represented by the first communications agent, that includes a mount descriptor that identifies a data object **in** a second data store in the second execution environment; detecting the mount descriptor; determining a first location in a first data store in the first execution environment; and creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store.

[0010] Further, a system for sharing a data object in a data store via a communication is described. The system includes a content manager component for receiving, by a first communications agent in a first execution environment according to a first communication protocol via a network from a second communications agent in a second execution environment, a first message, in a communication and addressed to a first user represented by the first communications agent, that includes a mount descriptor that identifies a data object in a second data store in the second execution environment. The system further includes a descriptor handler component for detecting the mount descriptor. The system still further includes a data store navigator component for determining a first location in a first data store in the first execution environment. The system additionally includes a mount service component for, creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store. The system also includes a processor, wherein at least one of the content manager component, the descriptor handler component, the data store navigator component, and the mount service component includes an instruction that is executed by the processor during operation of the system.

[0011] Methods and systems are described for sharing a data object in a data store via a communication. In one aspect, the method includes receiving, by a second communications agent that represents a second user, data object information that

Motion Offense Exhibit 2004

identifies a data object in a second data store in a second execution environment that includes the second communications agent. The method further includes creating a mount descriptor for accessing the data object from the second data store by a first data store in a first execution environment including a first communications agent that represents a first user. The method still further includes including, by the second communications agent, the mount descriptor in a first message, addressed to the first user. The method additionally includes sending, via a communications protocol via the network by the second communications agent, the first message for delivery to the first communications agent to create, based on the mount descriptor, a representation, of the data object in the first data storage, that when accessed includes accessing the data object from the second data store. Performing at least one the preceding actions comprising the method includes execution of an instruction by a processor.

[0012] Also, a system for sharing a data object in a data store via a communication is described that includes one or more processors and logic encoded in one or more tangible media for execution by the one or more processors that when executed is operable for receiving, by a second communications agent that represents a second user, data object information that identifies a data object in a second data store in a second execution environment that includes the second communications agent; creating a mount descriptor for accessing the data object from the second data store by a first data store in a first execution environment including a first communications agent that represents a first user; including, by the second communications agent, the mount descriptor in a first message, addressed to the first user; and sending, via a communications protocol via the network by the second communications agent, the first message for delivery to the first communications agent to create, based on the mount descriptor, a representation, of the data object in the first data storage, that when accessed includes accessing the data object from the second data store.

[0013] Further, a system for sharing a data object in a data store via a communication is described. The system includes a descriptor handler component for receiving, by a second communications agent that represents a second user, data object information

Motion Offense Exhibit 2004

that identifies a data object in a second data store in a second execution environment that includes the second communications agent. The system further includes a descriptor generator component for creating a mount descriptor for accessing the data object from the second data store by a first data store in a first execution environment including a first communications agent that represents a first user. The system still further includes a content manager component for including, by the second communications agent, the mount descriptor in a first message, addressed to the first user. The system additionally includes a com-out component for, sending, via a communications protocol via the network by the second communications agent, the first message for delivery to the first communications agent to create, based on the mount descriptor, a representation, of the data object in the first data storage, that when accessed includes accessing the data object from the second data store. The system also includes a processor, wherein at least one of the descriptor handler component, the descriptor generator component, the content manager component, and the com-out component includes an instruction that is executed by the processor during operation of the system.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] Objects and advantages of the present invention will become apparent to those skilled in the art upon reading this description in conjunction with the accompanying drawings, in which like reference numerals have been used to designate like or analogous elements, and in which:

[0015] FIG. **1** is a block diagram illustrating an exemplary hardware device included in and/or otherwise providing an execution environment in which the subject matter may be implemented;

[0016] FIG. **2A** is a flow diagram illustrating a method for sharing a data object in a data store via a communication according to an aspect of the subject matter described herein;

Motion Offense Exhibit 2004

[0017] FIG. **2B** is a flow diagram illustrating a method for sharing a data object in a data store via a communication according to an aspect of the subject matter described herein;

[0018] FIG. **3A** is a block diagram illustrating an arrangement of components for sharing a data object in a data store via a communication according to another aspect of the subject matter described herein;

[0019] FIG. **3B** is a block diagram illustrating an arrangement of components for sharing a data object in a data store via a communication according to another aspect of the subject matter described herein;

[0020] FIG. **4** is a block diagram illustrating an arrangement of components for sharing a data object in a data store via a communication according to another aspect of the subject matter described herein;

[0021] FIG. **5** is a network diagram illustrating a system for sharing a data object in a data store via a communication according to another aspect of the subject matter described herein;

[0022] FIG. **6A** is a diagram illustrating a user interface presented via a display according to another aspect of the subject matter described herein;

[0023] FIG. **6B** is a diagram illustrating a user interface presented via a display according to another aspect of the subject matter described herein;

[0024] FIG. **6C** is a diagram illustrating a user interface presented via a display according to another aspect of the subject matter described herein;

[0025] FIG. **6D** is a diagram illustrating a user interface presented via a display according to another aspect of the subject matter described herein;

[0026] FIG. **6E** is a diagram illustrating a user interface presented via a display according to another aspect of the subject matter described herein;

[0027] FIG. **6F** is a diagram illustrating a user interface presented via a display according to another aspect of the subject matter described herein;

[0028] FIG. **6G** is a diagram illustrating a user interface presented via a display according to another aspect of the subject matter described herein;

Motion Offense Exhibit 2004

[0029] Fig. **7** is a message flow diagram illustrating an exemplary data and execution flow for sharing a data object in a data store via a communication according to an aspect of the subject matter described herein;

[0030] Fig. **8A** illustrates an exemplary portion of a communication between an execution environment of a first node and an execution environment of a second node according to an aspect of the subject matter described herein; and

[0031] Fig. **8B** illustrates another exemplary portion of a communication between an execution environment of a first node and an execution environment of a second node according to an aspect of the subject matter described herein.

DETAILED DESCRIPTION

[0032] One or more aspects of the disclosure are described with reference to the drawings, wherein like reference numerals are generally utilized to refer to like elements throughout, and wherein the various structures are not necessarily drawn to scale. In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of one or more aspects of the disclosure. It may be evident, however, to one skilled in the art, that one or more aspects of the disclosure may be practiced with a lesser degree of these specific details. In other instances, well-known structures and devices are shown in block diagram form in order to facilitate describing one or more aspects of the disclosure.

[0033] Unless otherwise defined, all technical and scientific terms used herein have the same meaning as commonly understood by one of ordinary skill in the art to which this disclosure belongs. Although methods, components, and devices similar or equivalent to those described herein can be used in the practice or testing of the subject matter described herein, suitable methods, components, and devices are described below.

[0034] All publications, patent applications, patents, and other references mentioned herein are incorporated by reference in their entirety. In case of conflict, the present

Motion Offense Exhibit 2004

specification, including definitions, will control. In addition, the materials, methods, and examples are illustrative only and not intended to be limiting.

[0035] An exemplary device included in an execution environment that may be configured according to the subject matter is illustrated in FIG. **1**. An "execution environment", as used herein, is an arrangement of hardware and, in some aspects, software that may be further configured to include and/or otherwise host an arrangement of components for performing a method of the subject matter described herein. An execution environment includes and/or is otherwise provided by one or more devices. The execution environment is said to be the execution environment of the device and/or devices. An execution environment may be and/or may include a virtual execution environment including software components operating in a host execution environment. Exemplary devices included in and/or otherwise providing suitable execution environments for configuring according to the subject matter include personal computers, notebook computers, tablet computers, servers, portable electronic devices, handheld electronic devices, mobile devices, multiprocessor devices, distributed systems, consumer electronic devices, routers, communication servers, and/or any other suitable devices. Those skilled in the art will understand that the components illustrated in FIG. **1** are exemplary and may vary by particular execution environment.

[0036] FIG. **1** illustrates a hardware device **100** included in an execution environment **102**. FIG. **1** illustrates that execution environment **102** includes a processor **104**, such as one or more microprocessors; a physical processor memory **106** including storage locations identified by addresses in a physical memory address space of processor **104**; a persistent secondary storage **108**, such as one or more hard drives and/or flash storage media; an input device adapter **110**, such as a key or keypad hardware, a keyboard adapter, and/or a mouse adapter; an output device adapter **112**, such as a display and/or an audio adapter for presenting information to a user; a network interface component, illustrated by a network interface adapter **114**, for communicating via a network such as a LAN and/or WAN; and a communication mechanism that operatively couples elements **104-114**, illustrated as a bus **116**. Elements **104-114** may be

Motion Offense Exhibit 2004

operatively coupled by various means. Bus **116** may comprise any type of bus architecture, including a memory bus, a peripheral bus, a local bus, and/or a switching fabric.

[0037] As used herein a "processor" is an instruction execution machine, apparatus, or device. A processor may include one or more electrical, optical, and/or mechanical components that operate in interpreting and executing program instructions. Exemplary processors include one or more microprocessors, digital signal processors (DSPs), graphics processing units, application-specific integrated circuits (ASICs), optical or photonic processors, and/or field programmable gate arrays (FPGAs). Processor **104** may access machine code instructions and data via one or more memory address spaces in addition to the physical memory address space. A memory address space includes addresses identifying locations in a processor memory. The addresses in a memory address space are included in defining a processor memory. Processor **104** may have more than one processor memory. Thus, processor **104** may have more than one memory address space. Processor **104** may access a location in a processor memory by processing an address identifying the location. The processed address may be identified by an operand of a machine code instruction and/or may be identified by a register or other portion of processor **104**.

[0038] FIG. **1** illustrates a virtual processor memory **118** spanning at least part of physical processor memory **106** and may span at least part of persistent secondary storage **108**. Virtual memory addresses in a memory address space may be mapped to physical memory addresses identifying locations in physical processor memory **106**. An address space for identifying locations in a virtual processor memory is referred to as a virtual memory address space; its addresses are referred to as virtual memory addresses; and its processor memory is referred to as a virtual processor memory or virtual memory. The term "processor memory" may refer to physical processor memory, such as processor memory **106**, and/or may refer to virtual processor memory, such as virtual processor memory **118**, depending on the context in which the term is used.

Motion Offense Exhibit 2004

[0039] Physical processor memory **106** may include various types of memory technologies. Exemplary memory technologies include static random access memory (SRAM) and/or dynamic RAM (DRAM) including variants such as dual data rate synchronous DRAM (DDR SDRAM), error correcting code synchronous DRAM (ECC SDRAM), RAMBUS DRAM (RDRAM), and/or XDR™ DRAM. Physical processor memory **106** may include volatile memory as illustrated in the previous sentence and/or may include nonvolatile memory such as nonvolatile flash RAM (NVRAM) and/or ROM.

[0040] Persistent secondary storage **108** may include one or more flash memory storage devices, one or more hard disk drives, one or more magnetic disk drives, and/or one or more optical disk drives. Persistent secondary storage may include a removable data storage medium. The drives and their associated tangible computer readable storage media provide volatile and/or nonvolatile storage for computer-readable instructions, data structures, program components, and other data for execution environment **102**.

[0041] Execution environment **102** may include software components stored in persistent secondary storage **108**, in remote storage accessible via a network, and/or in a processor memory. FIG. **1** illustrates execution environment **102** including an operating system **120**, one or more applications **122**, and other program code and/or data components illustrated by other libraries and subsystems **124**. In an aspect, some or all software components may be stored in locations accessible to processor **104** in a shared memory address space shared by the software components. The software components accessed via the shared memory address space are stored in a shared processor memory defined by the shared memory address space. In another aspect, a first software component may be stored in one or more locations accessed by processor **104** in a first address space and a second software component may be stored in one or more locations accessed by processor **104** in a second address space. The first software component is stored in a first processor memory defined by the first address space and the second software component is stored in a second processor memory defined by the second address space.

Motion Offense Exhibit 2004

[0042] Software components typically include instructions executed by processor **104** in a computing context referred to as a "process". A process may include one or more "threads". A "thread" includes a sequence of instructions executed by processor **104** in a computing sub-context of a process. The terms "thread" and "process" may be used interchangeably herein when a process includes only one thread.

[0043] Execution environment **102** may receive user-provided information via one or more input devices illustrated by an input device **128**. Input device **128** provides input information to other components in execution environment **102** via input device adapter **110**. Execution environment **102** may include an input device adapter for a keyboard, a touch screen, a microphone, a joystick, a television receiver, a video camera, a still camera, a document scanner, a fax, a phone, a modem, a network interface adapter, and/or a pointing device, to name a few exemplary input devices.

[0044] Input device **128** included in execution environment **102** may be included in device **100** as FIG. **1** illustrates or may be external (not shown) to device **100**. Execution environment **102** may include one or more internal and/or external input devices. External input devices may be connected to device **100** via corresponding communication interfaces such as a serial port, a parallel port, and/or a universal serial bus (USB) port. Input device adapter **110** receives input and provides a representation to bus **116** to be received by processor **104**, physical processor memory **106**, and/or other components included in execution environment **102**.

[0045] An output device **130** in FIG. **1** exemplifies one or more output devices that may be included in and/or that may be external to and operatively coupled to device **100**. For example, output device **130** is illustrated connected to bus **116** via output device adapter **112**. Output device **130** may be a display device. Exemplary display devices include liquid crystal displays (LCDs), light emitting diode (LED) displays, and projectors. Output device **130** presents output of execution environment **102** to one or more users. In some embodiments, an input device may also include an output device. Examples include a phone, a joystick, and/or a touch screen. In addition to various types of display devices, exemplary output devices include printers, speakers, tactile

Motion Offense Exhibit 2004

output devices such as motion-producing devices, and other output devices producing sensory information detectable by a user. Sensory information detected by a user is referred herein to as "sensory input" with respect to the user.

[0046] A device included in and/or otherwise providing an execution environment may operate in a networked environment communicating with one or more devices via one or more network interface components. The terms "communication interface component" and "network interface component" are used interchangeably herein. FIG. **1** illustrates network interface adapter (NIA) **114** as a network interface component included in execution environment **102** to operatively couple device **100** to a network. A network interface component includes a network interface hardware (NIH) component and optionally a network interface software (NIS) component.

[0047] Exemplary network interface components include network interface controller components, network interface cards, network interface adapters, and line cards. A node may include one or more network interface components to interoperate with a wired network and/or a wireless network. Exemplary wireless networks include a BLUETOOTH network, a wireless 802.11 network, and/or a wireless telephony network (e.g., a cellular, PCS, CDMA, and/or GSM network). Exemplary network interface components for wired networks include Ethernet adapters, Token-ring adapters, FDDI adapters, asynchronous transfer mode (ATM) adapters, and modems of various types. Exemplary wired and/or wireless networks include various types of LANs, WANs, and/or personal area networks (PANs). Exemplary networks also include intranets and internets such as the Internet.

[0048] The terms "network node" and "node" in this document both refer to a device having a network interface component for operatively coupling the device to a network. Further, the terms "device" and "node" used herein may refer to one or more devices and nodes, respectively, providing and/or otherwise included in an execution environment unless clearly indicated otherwise.

[0049] The user-detectable outputs of a user interface are generically referred to herein as "user interface elements" or abbreviated as "UI elements". More specifically,

Motion Offense Exhibit 2004

visual outputs of a user interface are referred to herein as "visual interface elements". A visual interface element may be a visual output of a graphical user interface (GUI). Exemplary visual interface elements include windows, textboxes, sliders, list boxes, drop-down lists, spinners, various types of menus, toolbars, ribbons, combo boxes, tree views, grid views, navigation tabs, scrollbars, labels, tooltips, text in various fonts, balloons, dialog boxes, and various types of button controls including check boxes and radio buttons. An application interface may include one or more of the elements listed. Those skilled in the art will understand that this list is not exhaustive. The terms "visual representation", "visual output", and "visual interface element" are used interchangeably in this document. Other types of UI elements include audio outputs referred to as "audio interface elements", tactile outputs referred to as "tactile interface elements", and the like.

[0050] A visual output may be presented in a two-dimensional presentation where a location may be defined in a two-dimensional space having a vertical dimension and a horizontal dimension. A location in a horizontal dimension may be referenced according to an X-axis and a location in a vertical dimension may be referenced according to a Y-axis. In another aspect, a visual output may be presented in a three-dimensional presentation where a location may be defined in a three-dimensional space having a depth dimension in addition to a vertical dimension and a horizontal dimension. A location in a depth dimension may be identified according to a Z-axis. A visual output in a two-dimensional presentation may be presented as if a depth dimension existed allowing the visual output to overlie and/or underlie some or all of another visual output.

[0051] A "user interface (UI) element handler" component, as the term is used herein, includes a component of configured to send information representing a program entity for presenting a user-detectable representation of the program entity by an output device, such as a display. A "program entity" is an object included in and/or otherwise processed by an application or executable. The user-detectable representation is presented based on the sent information. Information that represents a program entity for presenting a user detectable representation of the program entity by an output

Motion Offense Exhibit 2004

device is referred to herein as "presentation information". Presentation information may include and/or may otherwise identify data in one or more formats. Exemplary formats include image formats such as raw pixel data, JPEG, video formats such as MP4, markup language data such as hypertext markup language (HTML) and other XML-based markup, a bit map, and/or instructions such as those defined by various script languages, byte code, and/or machine code. For example, a web page received by a user agent from a remote application provider may include HTML, ECMAScript, and/or byte code for presenting one or more UI elements included in a user interface of the remote application. Components configured to send information representing one or more program entities for presenting particular types of output by particular types of output devices include visual interface element handler components, audio interface element handler components, tactile interface element handler components, and the like.

[0052] A representation of a program entity may be stored and/or otherwise maintained in a presentation space. As used in this document, the term "presentation space" refers to a storage region allocated and/or otherwise provided for storing presentation information, which may include audio, visual, tactile, and/or other sensory data for presentation by and/or on an output device. For example, a buffer for storing an image and/or text string may be a presentation space as sensory information for a user. A presentation space may be physically and/or logically contiguous or non-contiguous. A presentation space may have a virtual as well as a physical representation. A presentation space may include a storage location in a processor memory, secondary storage, a memory of an output adapter device, and/or a storage medium of an output device. A screen of a display, for example, is a presentation space.

[0053] As used herein, the terms "program" and "executable" refer to any data representation that may be translated into a set of machine code instructions and may optionally include associated program data. The terms are used interchangeably herein. Program representations other than machine code include object code, byte code, and source code. Object code includes a set of instructions and/or data elements that either

Motion Offense Exhibit 2004

are prepared for linking prior to loading or are loaded into an execution environment. When in an execution environment, object code may include references resolved by a linker and/or may include one or more unresolved references. The context in which this term is used will make clear the state of the object code when it is relevant. This definition can include machine code and virtual machine code, such as Java™ byte code. As used herein, the terms "application", and "service" may be realized in one or more executables and/or in one or more hardware components. The terms are used interchangeably herein.

[0054] As used herein, the term "network protocol" refers to a formal set of rules, conventions and data structures that governs how computers and other network devices exchange information over a network. The rules, conventions, and data structures are said to be specified or defined in a specification and/or schema.

[0055] The term "network path" as used herein refers to a sequence of nodes in a network that are communicatively coupled for transmitting data in one or more data units of a specified network protocol between a pair of nodes in the network.

[0056] A "data unit", as the term is used herein, is a data entity specified according to a network protocol for transmitting data contained in and/or otherwise identified by the data entity. A data unit is transmitted between a pair of nodes in a network path to send the data in and/or otherwise identified by the data unit from a source node to a destination node that includes a protocol endpoint of the network protocol. A network protocol explicitly and/or implicitly specifies and/or otherwise identifies a schema that defines one or more of a format for a valid data unit and a vocabulary for content of a valid data unit. One example of a data unit is an Internet Protocol (IP) packet. The Internet Protocol defines a format for an IP packet that defines a header for identifying a destination address that identifies a destination node and a payload portion for including a representation of data to be delivered to the identified destination node. The terms "data unit", "data packet", and "packet" are used interchangeably herein. One or more data units of a first network protocol may transmit a message of second network protocol. For example, one or more data units of the IP protocol may include a TCP

Motion Offense Exhibit 2004

message. In another example, one or more TCP data units may transmit an HTTP message.

[0057] How data is packaged in one more data units for a network protocol may vary as the data traverses a network path from a source node to a destination node. Data may be transmitted in a single data unit between two consecutive nodes in a network path. Additionally, data may be exchanged between a pair of consecutive nodes in several data units each including a portion of the data. Data received in a single data unit by a node in a network path may be split into portions included in several respective data units for transmitting to a next node in the network path. Portions of data received in several data units may be combined into a single data unit for transmitting by a node in a network path. For purposes of describing the subject matter, a data unit in which data is received by a node is referred to as a different data unit than a data unit in which the data is forwarded by the node.

[0058] A "protocol address", as the term is used herein, for a network protocol is an identifier of a protocol endpoint that may be represented in a data unit of the protocol. For example 192.168.1.1 is an IP protocol address represented in a human readable format that may be included in an address portion of an IP header to identify a source and/or a destination IP protocol endpoint. A protocol address differs from a symbolic identifier, defined below, in that a symbolic identifier, with respect to a network protocol, maps to a protocol address. Thus, "www. mynode.com" may be a symbolic identifier for a node in a network when mapped to the protocol address 192.168.1.1. An identifier may be both a symbolic identifier and a protocol address depending on its role with respect to its use for a particular network protocol.

[0059] A node in a pair of nodes in a network path at one end of the sequence of nodes in the network path and/or the other end is referred to herein as a "path end node". Note that a node may have two NICs with one NIC at each end of a network path. A network path may be included as a portion of another network path that communicatively couples a same pair of nodes. Data may be transmitted via the sequence of nodes in a network path between path end nodes communicatively

Motion Offense Exhibit 2004

coupled via the network path. Data may be transmitted in one or both directions depending on an ordering of the nodes in the sequence.

[0060] "Path information" is any information that identifies a network path and/or hop path for data transmitted via one or more specified network protocols. Path information may be identified by identifying network interfaces, NICs, nodes, and/or hops included in a network path. "Address information" is any information that identifies a protocol address that, for a network protocol, identifies a protocol endpoint. Address information may identify a unicast protocol address for a network protocol. In identifying a protocol endpoint, a protocol address identifies a node and a network interface. Those skilled in the art will understand upon reading the descriptions herein that the subject matter disclosed herein is not restricted to the network protocols described and/or their corresponding OSI layers. For ease of illustration, the subject matter is described in terms of protocols that correspond to OSI layer three, also referred to as network layer protocols, in general. Particular descriptions are based on versions of the Internet Protocol (IP). Address information may identify one or more protocol addresses. Exemplary protocol addresses include IP addresses, IPX addresses, DECNet addresses, VINES Internet Protocol addresses, and Datagram Delivery Protocol (DDP) addresses.

[0061] An "interaction", as the term is used herein, refers to any activity including a user and an object where the object is a source of sensory data detected by the user. In an interaction the user directs attention to the object. An interaction may also include the object as a target of input from the user. The input from the user may be provided intentionally or unintentionally by the user. For example, a rock being held in the hand of a user is a target of input, both tactile and energy input, from the user. A portable electronic device is a type of object. In another example, a user looking at a portable electronic device is receiving sensory data from the portable electronic device whether the device is presenting an output via an output device or not. The user manipulating an input component of the portable electronic device exemplifies the device, as an input target, receiving input from the user. Note that the user in providing input is detecting

Motion Offense Exhibit 2004

sensory information from the portable electronic device provided that the user directs sufficient attention to be aware of the sensory information and provided that no disabilities prevent the user from processing the sensory information. An interaction may include an input from the user that is detected and/or otherwise sensed by the device. An interaction may include sensory information that is detected by a user included in the interaction that is presented by an output device included in the interaction.

[0062] As used herein "interaction information" refers to any information that identifies an interaction and/or otherwise provides data about an interaction between a user and an object, such as a portable electronic device. Exemplary interaction information may identify a user input for the object, a user-detectable output presented by an output device of the object, a user-detectable attribute of the object, an operation performed by the object in response to a user, an operation performed by the object to present and/or otherwise produce a user-detectable output, and/or a measure of interaction. The term "operational component" of a device, as used herein, refers to a component included in performing an operation by the device.

[0063] Interaction information for one object may include and/or otherwise identify interaction information for another object. For example, a motion detector may detect user's head turn in the direction of a display of a portable electronic device. Interaction information identifying the user's head is facing the display may be received and/or used as interaction information for the portable electronic device indicating the user is receiving visual input from the display. The interaction information may serve to indicate a lack of user interaction with one or more other objects in directions from the user different than the detected direction, such as a person approaching the user from behind the user. Thus the interaction information may serve as interaction information for one or more different objects.

[0064] As used herein, any reference to an entity "in" an association is equivalent to describing the object as "identified" by the association, unless explicitly indicated otherwise.

Motion Offense Exhibit 2004

[0065] As used herein, the term "communication" refers to information including a message sent and/or for sending via a network between communicants. A message may include text data, audio data, and/or image data. The term "communicant" as used herein refers to a user included in a communication as a sender and/or a receiver of the information. A communicant is represented by a "communications agent" configured to operate in an execution environment to send data to and/or receive data from another communications agent, on behalf of the represented communicant, according to a communications protocol via network. A communications protocol defines and/or otherwise identifies an address space including communications addresses for delivering data sent in a communication from one communications agent to another. A communications protocol is a type of network protocol.

[0066] The term "communicant alias" as used herein refers to an identifier of a communicant in a communication where the communicant alias is not a communications address included in an address space of a communications protocol for sending and/or receiving data in the communication.

[0067] The term "attachment" as used herein refers to a portion of a communication that includes data from one communicant to another other than data in the message portion. A resource sent as an attachment is data that is typically not presented "inline" or in a message included in a message portion of a communication. Email attachments are perhaps the most widely known attachments included in communications. An email attachment is a file or other resource sent along with an email in a portion of the email separate from a message portion. As defined, other types of text communications as well as voice and video communications may include mount descriptor portions. A communication may include one or more resources as one or more attachments.

[0068] The terms "contactor" and "contactee" identify roles in a communication. A "contactor" provides information for identifying a "contactee" in a communication. A contactee may be included in a communication by a contactor and/or the contactor's communications agent. A contactor and/or a contactee are roles of a user or communicant of a communications agent and may be a living being, a node, a

Motion Offense Exhibit 2004

component, and/or an application. Both a contactor and a contactee are communicants in a communication.

[0069] The term "data object" as used herein refers to an entity, identifiable within a data store, for storing and/or accessing data. Exemplary data stores include file systems, directory service including DNS and LDAP, and data bases including relational databases and hierarchical databases, to name a few examples. Exemplary data objects include files and folders in a file system. Accessing a data object includes one or more of accessing data in the data object and accessing metadata, maintained by the data store, about the data object.

[0070] FIG. **3A** illustrates an arrangement of components in a system that operates in an execution environment, such as execution environment **102** in FIG. **1**. The arrangement of components in the system operates to perform the method illustrated in FIG. **2A**. The system illustrated includes a content manager component **302**, a descriptor handler component **304**, a data store navigator component **306**, and a mount service component **308**. A suitable execution environment includes a processor, such as processor **104**, to process an instruction in at least one of a content manager component, a descriptor handler component, and a data store navigator component, and a mount service component. FIG. **3B** illustrates an arrangement of components in a system that operates to perform the method illustrated in FIG. **2B**. The system illustrated includes a descriptor handler component **304**, a descriptor generator component **314**, a content manager component **302**, and a com-out component **318**. A suitable execution environment includes a processor, such as processor **104**, to process an instruction in at least one of a descriptor handler component, a descriptor generator component, a content manager component, and a com-out component.

[0071] Some components, illustrated in the drawings are identified by numbers with an alphanumeric suffix. A component may be referred to generically in the singular or the plural by dropping a suffix of a portion thereof of the component's identifier. For example, window UI elements, such as a window UI element **602a** in FIG. **6A**, a window UI element **602a** in FIG. **6A**, and their adaptations and analogs; are referred to herein

Motion Offense Exhibit 2004

generically as a window UI element **602** or window UI elements **602** when describing more than one. Other components identified with an alphanumeric suffix may be referred to generically or as a group in a similar manner.

[0072] The arrangement of components illustrated in FIG. **4** may be adapted to operate in a number of execution environments to perform the method illustrated in FIG. **2A** and/or the method illustrated in FIG. **2B**. FIG. **4** is a block diagram illustrating the components of FIG. **3A**, FIG. **3B**, and/or analogs of the components of FIG. **3A** and FIG. **3B**, respectively adapted to operate in an execution environment **401** that includes and/or otherwise is provided by one or more nodes. FIG. **1** illustrates key components of an exemplary device that may at least partially provide and/or otherwise be included in an execution environment. The components illustrated in FIG. **4** may be included in or otherwise combined with the components of FIG. **1** to create a variety of arrangements of components according to the subject matter described herein.

[0073] As stated, the various adaptations and analogs of the arrangement in Fig. **3A** as well as the various adaptations and analogs of the arrangement in Fig. **3B** illustrated and described herein are not exhaustive. For example, those skilled in the art will see based on the description herein that arrangements of components for performing the method illustrated in Fig. **2A** and the method illustrated in Fig. **2B** may each be distributed across more than one node and/or execution environment. For example, such arrangements may operate at least partially in a browser in a one execution environment and at least partially in a server in another execution environment interoperating via a network.

[0074] Fig. **5** illustrates a first node **502** and a second node as exemplary devices that each may be included in and/or otherwise adapted for providing an instance, adaptation, and/or analog of execution environment **401** in Fig. **4**. As illustrated in Fig. **5**, first node **502** and a second node **504** are operatively coupled to a network **506** via respective network interface components enabling execution environment **401** of first node **502** and execution environment **401** of second node **504** to exchange data in a communication on behalf of communicants represented by the respective nodes.

Motion Offense Exhibit 2004

[0075] Fig. **4** illustrates a communications agent **403** including an adaptation of the arrangement of components in Fig. **3A** and an adaptation of the arrangement of components in Fig. **3B**. A first communications agent **403** may operate, in execution environment **401** of first node **502**, on behalf of a first communicant to communicate with a second communicant represented by a second communications agent **403** operating in execution environment **401** of second node **504**.

[0076] An execution environment **401** is illustrated in Fig. **4** hosting communications agent **403**. Components in Fig. **4** are referenced for illustrative purposes in describing communications agents operating in an execution environment of first node **502** and an execution environment **401** of second node **504**. Exemplary communications agents include email clients, phone clients including Voice over Internet Protocol (VoIP) clients, instant messaging clients, short message service (SMS) clients, multimedia message service (MMS clients), multi-media communications clients including video phone clients, and other data transfer agents.

[0077] Communications agents **403** such as illustrated in Fig. **4**, respectively operating in execution environment **401** of first node **502** and in execution environment **401** of second node **504** in Fig. **5** may interoperate via respective network stacks **405**. Communications agents **403** may communicate via one or more communications protocols. Fig. **4** illustrates communications protocol component **407** exemplifying a subsystem for exchanging data via network **506** according to one or more communications protocols, such as simple mail transfer protocol (SMTP), an instant messaging protocol, and/or a real-time voice and/or video protocol. A communication between communications agents **403** in execution environment **401** of first node **502** and execution environment **401** of second node **504** may include more than one type of data and may use one or more communications protocols in exchanging one or more types of data via network **506**.

[0078] Instances, adaptations, and/or analogs of communications agent **403** in Fig. **4**, in execution environment **401** of first node **502** and in execution environment **401** of second node **504**, respectively, may communicate via discrete messages, a

Motion Offense Exhibit 2004

request/reply protocol, a data streaming protocol, a session and/or connection-oriented protocol, a connectionless protocol, a real-time communications protocol, an asynchronous communication, a store and forward communications protocol, a reliable delivery communications protocol, a best-effort delivery communications protocol, and/or a secure protocol, to name a few communications options.

[0079] Fig. **4** illustrates communications agent **403** including a content manager component **402**. Content manager component **402** may interoperate with communications protocol layer component **407** and/or network stack **405** to receive data in one or more communications via network **506** with another communications agent in another node. Content manager component **402**, in FIG. **4**, is operatively coupled, via com-in component **411**, to communications protocol component **407** and operates to receive the data from the other node.

[0080] Data received in a communication may include one or more data portions having respective content types. Exemplary content types include plain text, markup such as hypertext markup language (HTML), audio data, image data, and/or executable data. Executable data may include script instruction(s), byte code, and/or machine code. In Fig. **4**, communications agent **403** includes one or more content handler components **413** to process data received according to its content type. A data type may be identified by a MIME type identifier. Exemplary content handler components **413** include text/html content handler components for processing HTML representations; application/xmpp-xml content handler components for processing extensible messaging and presence protocol (XMPP) streams including presence tuples, instant messages, and audio content handlers including and/or configured to retrieve suitable codices; video content handler components for processing video representations of various types; and still image data content handler components for processing various image data representations.

[0081] Content handler component(s) **413** process received data representations and may provide transformed data from the representations to one or more user interface element handler components **415**. One or more user interface element handler

Motion Offense Exhibit 2004

components **415** are illustrated in a presentation controller component **417** in Fig. **4**. Presentation controller component **417** may manage visual, audio, and other types of output for its including application as well as receive and route detected user and other inputs to components and extensions of its including application, communications agent **403**. A user interface element handler component **415** may be adapted to operate at least partially in a content handler component **413** such as a text/html content handler component and/or a script content handler component. Additionally or alternatively, a user interface element handler component in an execution environment **401** may be received in a communication. For example, a communication, such as an email, may include an HTML content type portion and a script content type portion.

[0082] Figs. **6A-H** illustrate various user interfaces, some of which include window UI elements **602** presentable in a presentation space of a display device, such as output device **130** in Fig. **1**. A window UI element **602** may include a contactor UI element **604** for presenting an identifier of a communicant in the role of a contactor in a communication represented by the window UI element **602**. A window UI element **602** may also include a contactee UI element **606** for presenting one or more contactee identifier(s) identifying one or more communicants in the role of contactee(s) in the communication. A presentation space **608** may be included in each window UI element **602** for presenting a user message UI element **610** that my present a user message from one user to another. A presentation space **608** may also be provided for presenting one or more UI elements that correspond to user inputs for exchanging data in and/or otherwise managing a communication.

[0083] Data to send in a communication to a communications agent in execution environment **401** of second node **504** may be received by one or more content handler component(s) **413** operating in execution environment **401** of first node **502** to transform the data into one or more data representations suitable for transmitting in the communication and/or suitable for processing by the communications agent in execution environment **401** of second node **504**. The one or more data representations may be provided to content manager component **402** for sending in the communication

Motion Offense Exhibit 2004

to execution environment **401** of second node **504**. Content manager component **402** may package the one or more data representations in a message formatted according to a communications protocol of the communications agent. Communications protocol component **407** may send the data according to the specification(s) of the communications protocol. Content manager component **402** may alternatively or additionally encode and/or otherwise transform one or more of the data representations so that the data may be sent in a data stream such as voice stream and/or a video stream in the communication to a communications agent via a network.

[0084] Content manager component **402** operating in execution environment **401** included in and/or otherwise provided by a node, such as execution environment **401** of first node **502**, may provide the packaged, encoded, and/or transformed data to communications protocol component **407** via a com-out component **418**. Com-out component **418** as described above operatively couples communications agent **402** to communications protocol component **407** according to an interface provided by communications protocol component **407** to send data in a communication according to a communications protocol. Communications protocol component **407** may further package and/or otherwise transform the data to send via network stack **405** for delivery via network **506** to another node, such as second node **504**.

[0085] As described above, a communications protocol may operate via one or more nodes in a network in a network path communicatively coupling a first node and a second node. Exemplary path nodes include mail relay nodes, phone switch nodes, and proxy nodes such as instant messaging proxies for communicating through firewalls. As indicated, proxy node **508** may operate as a path node.

[0086] A contactee is identified for sending data in a communication. A contactee may be represented by a communications address in an address space of a communications protocol. In one aspect, information identifying a communications address may be received from a communicant of a communications agent in an execution environment. In Fig. **4**, presentation controller **417** and/or a UI element handler **415**, presenting and/or managing interaction with contactor UI element **604a** in Fig. **6A** for execution

Motion Offense Exhibit 2004

environment **401** of first node **502**, may receive a contactor alias in response to a user input corresponding to UI element handler **415**. The user/communicant of execution environment **401** of first node **502** may enter a contactor alias, such as "William", via a keyboard and/or select a predefined communicant alias presented in a selection UI control element via a UI element handler component **415**. The user input may be detected by input driver **421**. Corresponding input information may be routed to presentation controller **417** by GUI subsystem **423**. GUI subsystem **423** operates to send presentation information to a display device via a graphics subsystem **425**. Communication agent **403** may identify a communication address associated with the contactor alias, "William". Other communications address, such as for one or more contactees, may be received similarly and/or in any suitable manner. In an aspect, one or more contactor and/or contactee identifiers may be determined automatically.

[0087] Data may be sent in a communication according to a form or type of the communication and/or other attribute of the communication such as a security attribute, the amount of data to be set, a priority setting, a task setting, and the like. Some forms of communication do not require a session and/or a connection between nodes representing communicants in the communication in order to send data between and/or among the nodes, while other forms of communication do. An email and/or instant message may use a store and forward model of delivery.

[0088] Data may be sent in a communication in response to a communicant input. In FIG. **6A** a contactor may provide an input corresponding to reply UI element **612a** in Fig. **6A**. The input may be received by presentation controller component **417** and/or one or more UI element handlers **415** corresponding to reply UI element **612a**. In response to detecting the input, presentation controller component **417** may provide data to be sent in the communication to one or more content handler components **413** according to the content type(s) of the data to be sent.

[0089] The one or more content handler components **413** may encode, format, and/or otherwise transform the data for sending in a message, such as an email message. The one or more content handler components **413** may provide data to be sent to content

Motion Offense Exhibit 2004

manager **402**, instructing content manager component **402** to send the data in the communication for delivery to a communications agent in execution environment **401** of second node **504**. Content manager component **402** interoperating with com-out component **418** may further format and/or transform the data for sending in the communication according to a communications protocol, for example according to an email communications protocol, by communications protocol component **407**. Communications protocol component **407** may send the communication for delivery to the communications agent in execution environment **401** of second node **504** via network **506**.

[0090] One or more path nodes may relay data sent in the communication between execution environment **401** of first node **502** and execution environment **401** of second node **504**. A path node may determine a next node and/or a network interface in a network path communicatively coupling execution environment **401** of first node **502** and execution environment **401** of second node **504** for exchanging data in a communication between communication agents **403** in the respective execution environments **401** of first node **502** and second node **504**.

[0091] For session-oriented and/or connection-oriented communication a session and/or connection may be established if a session/connection has not already been established. Data may be sent for delivery to a communications agent identified based on a contactee communications address during session and/or connection setup. For example, for a voice communication a voice communication may be established via a session initiation protocol. Communications protocol component **407** may operate according to the session initiation protocol specifications. Communications protocol component **407** operating in execution environment **401** of execution environment **401** of first node **502** may locate a communications agent by communicating with one or more nodes in network **506** according to the session initiation protocol. Communications protocol component **407** may locate a communications agent in execution environment **401** of second node **504**, based on a communications address for the contactee.

Motion Offense Exhibit 2004

[0092] Once a communication session is established, such as a voice session, data may be sent according to the session communications protocol, such as RTP. Data may be sent according to the session initiation protocol in the communication for managing the voice communication session and/or for exchanging text, image, and/or other data outside of the voice session. One or more path nodes may be included in session and/or connection setup. Alternatively or additionally, one or more path nodes may be included in a network path in a session and/or connection.

[0093] With reference to FIG. **2A**, a block **202** illustrates that the method includes receiving, by a first communications agent in a first execution environment according to a first communication protocol via a network from a second communications agent in a second execution environment, a first message, in a communication and addressed to a first user represented by the first communications agent, that includes a mount descriptor that identifies a data object in a second data store in the second execution environment. Accordingly, a system for sharing a data object in a data store via a communication includes means for receiving, by a first communications agent in a first execution environment according to a first communication protocol via a network from a second communications agent in a second execution environment, a first message, in a communication and addressed to a first user represented by the first communications agent, that includes a mount descriptor that identifies a data object in a second data store in the second execution environment. For example, the arrangement in FIG. **3A**, includes content manager component **302** that is operable for receiving, by a first communications agent in a first execution environment according to a first communication protocol via a network from a second communications agent in a second execution environment, a first message, in a communication and addressed to a first user represented by the first communications agent, that includes a mount descriptor that identifies a data object in a second data store in the second execution environment. FIG. **4** illustrates content manager component **402** as an adaptation and/or analog of the content manager component **302** in FIG. **3A**. One or more content manager components **402** operate in an execution environment **401**. The system for

Motion Offense Exhibit 2004

sharing a data object in a data store via a communication includes one or more processors and logic encoded in one or more tangible media for execution by the one or more processors that when executed is operable for receiving, by a first communications agent in a first execution environment according to a first communication protocol via a network from a second communications agent in a second execution environment, a first message, in a communication and addressed to a first user represented by the first communications agent, that includes a mount descriptor that identifies a data object in a second data store in the second execution environment.

[0094] FIG. **7** includes a data flow message that illustrates a first message **702**, including a mount descriptor, received via network **506** by execution environment **401** including and/or otherwise provided by first node **502**. First message **702** is sent via a communications protocol by a communications agent **403** in execution environment **401** including and/or otherwise provided by second node **504**.

[0095] First message **702** is may be addressed to a user of execution environment **401** of first node **502** by a communications address, that identifies the user, from an address space of the communications protocol. First message may be from a user, identified by a communications address in the address space of the communications protocol and represented by communications agent **403** in of execution environment **401** of second node **504**. The communications protocol may include a protocol specified for exchanging at least one of an email, an instant message, a short messaging service message (SMS), a multi-media message (MMS), an audio message, a video message, and an attachment. First message **702** may include a user readable message from the user of execution environment **401** of second node **504**. Receiving first message **702** may include detecting the mount descriptor in first message **702** along with the user readable message.

[0096] As described above, content manager component **402** operating in execution environment **401** of first node **502** may interoperate with communications protocol layer component **407** and/or network stack **405** to receive data in one or more

Motion Offense Exhibit 2004

communications via network **506** with another communications agent in another node. A communication may be direct and/or indirect, such via an email relay. Content manager component **402** in FIG. **4** operatively couples one or more content handler components **413** with network **506**, via com-in component **411**, communications protocol component **407**, and/or network stack **405** to receive and process the data from the other execution environment of the other node. Thus, content manager component **402** operating in execution environment **401** of first node **502** may receive data, such as sent via first message **702**, which includes a mount descriptor. FIG. **8A** includes an exemplary content portion **800a** of an email message. Content portion **800a** includes a user readable message portion **802a** and a mount descriptor portion **804a** described in more detail below. The received communication may include a message addressed to a user represented by communications agent **403** in execution environment **401** of first node **502**.

[0097] Returning to FIG. **2A**, a block **204** illustrates that the method further includes detecting the mount descriptor. Accordingly, a system for sharing a data object in a data store via a communication includes means for detecting the mount descriptor. For example, the arrangement in FIG. **3A**, includes descriptor handler component **304** that is operable for detecting the mount descriptor. FIG. **4** illustrates descriptor handler component **404** as an adaptation and/or analog of descriptor handler component **304** in FIG. **3A**. One or more descriptor handler components **404** operate in an execution environment **401**. The system for sharing a data object in a data store via a communication includes one or more processors and logic encoded in one or more tangible media for execution by the one or more processors that when executed is operable for detecting the mount descriptor. In FIG. **4**, descriptor handler component **404** is illustrated as a component of communications agent **403**.

[0098] FIG. **7** includes a data flow that illustrates a detect descriptor data flow **704** that may be internal to execution environment **401** of first node **502** and/or that may include a message received via network **506** by first node **502**. In an aspect, detect descriptor data flow **704** may illustrate a data flow to descriptor handler component **404** operating

Motion Offense Exhibit 2004

in execution environment **401** of first node **502**. A mount descriptor may be identified to and/or otherwise provided to descriptor handler component **404**. Content handler component **402**, operating as described with respect to block **202** of FIG. **2A**, may identify and/or otherwise provide a mount descriptor to a descriptor handler component **404**.

[0099] Content manager component **402** may provide some or all of the data in a received message, such as first message **702**, to descriptor handler component **404**. Descriptor handler component **404** may identify and/or otherwise detect a mount descriptor in the message sent from the execution environment **401** of second node **504**. Content manager component **402** may interoperate with descriptor handler component **404** in identifying the mount descriptor. The mount descriptor may be identified for processing by descriptor handler component **404** by content manager component **402** according to a schema for the message. For example, a portion of an email message may include a MIME-type identifier that identifies a corresponding part of the email as a mount descriptor and/or otherwise identifies the part for routing to descriptor handler component **404** in communications agent **403** in execution environment **401** of first node **502**.

[0100] Returning to FIG. **2A**, a block **206** illustrates that the method yet further includes determining a first location in a first data store in the first execution environment. Accordingly, a system for sharing a data object in a data store via a communication includes means for determining a first location in a first data store in the first execution environment. For example, the arrangement in FIG. **3A**, includes data store navigator component **306** that is operable for determining a first location in a first data store in the first execution environment. FIG. **4** illustrate data store navigator components **406** as an adaptations and/or analogs of data store navigator component **306** in FIG. **3A**. One or more data store navigator components **406** operate in an execution environment **401**. The system for sharing a data object in a data store via a communication includes one or more processors and logic encoded in one or more tangible media for execution by the one or more processors that when executed is operable for determining a first

Motion Offense Exhibit 2004

location in a first data store in the first execution environment. In FIG. **4**, a data store navigator component **406** is illustrated as a component of application **403**.

[0101] FIG. **7** includes a data flow message that illustrates a locate data flow **706** that may be internal to execution environment **401** of first node **502** and/or that may include a message received via network **506** by first node **502**. Locate data flow **706**, in an aspect, illustrates a data flow to data store navigator component **406** operating in execution environment **401** of first node **502**. Data store navigator component **406** may be invoked automatically in response to detecting a mount descriptor. In another aspect, data store navigator component **406** may be invoked in response to a user input.

[0102] FIG. **6A** illustrates a view window UI element **602a** as an exemplary user interface that may be presented by communications agent **403** operating in execution environment **401** of first node **502** to present data received in a communication, such as first message **702**, from execution environment **401** of second node **504**. FIG. **6A** illustrates a contactor UI element **604a**, a contactee UI element **606a**, and a presentation space **608a** of the view window UI element **602a**. Presentation space **608a** includes a user message UI element **610a**. The various UI elements may be presented by one or more UI element handler components **415** in communications agent **403** in execution environment **401** of first node **502**. A reply UI element **612a** and a reply all UI element **614a** illustrate UI elements in presentation space **608a** presented to allow a user of the execution environment **401** of first node **502** to respond and/or otherwise instruct execution environment **401** of first node **502** to perform an operation based on the message received in the communication. FIG. **6A** illustrates an attached mounts UI element **616a** identifying, based on the mount descriptor in first message **702**, a "Music" folder as a mountable folder from a data store of execution environment **401** of second node **504**.

[0103] In an aspect, determining a location in a data store for creating a representation of a data object may include presenting a navigation UI element for navigating the data store. The location may be determined in response to detecting a user input identifying the location. View window UI element **602a** in FIG. **6A** includes a mount UI element

Motion Offense Exhibit 2004

618a for receiving user input to display a navigation window UI element **620a** for identifying a location in a data store of execution environment **401** of first node **502** to mount and/or otherwise attach the remote "Music" folder identified in the mount descriptor. In response to an input directed to mount UI element **618a**, a corresponding UI element handler **415** may interoperate with data store navigator component **406** to present navigation window UI element **620a** that includes a presentation space **622a** for presenting a user detectable representation of one or more locations in a data store of execution environment **401** of first node **502**.

[0104] FIG. **6A** illustrates a mount UI element **624a** allowing a user to provide user input(s) to instruct communications agent **403** in execution environment **401** of first node **502** to request that a data object identified in the mount descriptor be attached to a location, identified in navigation window UI element **602b**, in the data store in execution environment **401** of first node **502**.

[0105] In another aspect, files and/or other data entities identified in a mount descriptor may be illustrated in a manner similar to attachments rather than in a separate UI element as in FIG. **6A**. Creating a representation of a data object may include receiving an indication by a communications agent to access the data object identified by a mount descriptor. In response, a location in a data store for creating the representation may be determined automatically. Further, in response to receiving the indication the representation may be created automatically at the location. Still further, in response to receiving the indication, the data object may be accessed via the network based on the mount descriptor.

[0106] In response to detecting the mount descriptor in first message **702**, execution environment **401** of first node **502** may present a view window UI element **602b** in FIG. **6B** including data received in the first message **702** to a user of execution environment **401** of first node **502** as described with respect to FIG. **6A**. Context menu **626b** provides UI elements to allow the user to instruct execution environment **401** of first node **502** to retrieve a data object, in response to an input corresponding to "Open" menu UI element **626b1**, identified in a mount descriptor received from execution environment

Motion Offense Exhibit 2004

401 of second node **504**. Context menu **626b** provides UI elements to allow the user to instruct execution environment **401** of first node **502** to attach and/or otherwise mount a representation of a data object, in response to an input corresponding to "Save..." menu UI element **626b2**, identified in a mount descriptor at a location in the data store of execution environment **401** of first node **502**.

[0107] Alternatively or additionally, a location for creating representation of a data object may be determined by detecting user input included in performing a drag and drop operation on a representation of the data object presented via an output device. The drag and drop operation may be performed to attach a data object identified in a received mount descriptor to create a representation of the data object at a location in a data store. FIG. **6B** includes a pointer UI element **628b** dragging a representation of the "Music" folder from remote attachments UI element **630b** in view window UI element **602b**. The representation of the "Music" folder is shown over a listing pane **632b**, in a navigation window UI element **634b**, showing data entities in the data store of execution environment **401** of first node **502** at a location identified in a path UI element **636b**.

[0108] FIG. **6C** illustrates another exemplary user interface that execution environment **401** of first node **502** may present to the user to allow the user to instruct execution environment **401** of first node **502** to mount and/or otherwise attach a representation of a data object identified in a mount descriptor to a location in a data store of execution environment **401** of first node **502**. Auto-response window **602c** may be presented in response to a mount descriptor received in a communication without a user message. Rather than a UI element for displaying a user message, a remote data object pane UI element **638c** may be presented that identifies data objects located and/or otherwise identified by execution environment **401** of second node **504** in a mount descriptor received by execution environment **401** of first node **502**. UI elements in presentation space **608c** are illustrated that allow a user to instruct execution environment **401** of first node **502** to retrieve a data object, in response to an input corresponding to "Open" UI element **640c**, identified in a mount descriptor received from execution environment **401** of second node **504**. Mount UI element **642c** allows the user to instruct execution

Motion Offense Exhibit 2004

environment **401** of first node **502** to attach and/or otherwise mount a representation of a data object, identified in a mount descriptor at a location in the data store of execution environment **401** of first node **502.**

[0109] Returning to FIG. **2A**, a block **208** illustrates that the method yet further includes creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store. Accordingly, a system for sharing a data object in a data store via a communication includes means for creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store. For example, the arrangement in FIG. **3A**, includes mount service component **308** that is operable for creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store. FIG. **4** illustrates mount service component **408** as an adaptation and/or analog of mount service component **308** in FIG. **3A**. One or more mount service components **408** operate in an execution environment **401**. The system for sharing a data object in a data store via a communication includes one or more processors and logic encoded in one or more tangible media for execution by the one or more processors that when executed is operable for creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store. In FIG. **4**, a mount service component **408** is illustrated as a component of data store navigator component **406**.

[0110] FIG. **7** includes data flow message that illustrates an attach data flow **708** that may be internal to execution environment **401** of first node **502** and/or that may include a message received via network **506** by execution environment **401** of first node **502**. Attach data flow **708** illustrates a data flow to mount service component **408** operating in execution environment **401** of first node **502**. Mount service component **408** may be invoked automatically in response to detecting a mount descriptor in an aspect. In

Motion Offense Exhibit 2004

another aspect, mount service component **408** may be invoked in response to a user input. A representation of a data object identified in a mount descriptor may be created automatically in response to detecting the mount descriptor and/or detecting a user input associated with the message in which the mount descriptor was received.

[0111] Once a location is identified, in a data store of an execution environment, for attaching and/or otherwise mounting a representation of a data object in a data store in another execution environment; any of a variety means available to the execution environment mounting the data object may be utilized.

[0112] FIG. **8A** illustrates a mount descriptor identifying alternative URLs for accessing the "Music" folder in the data store in execution environment **401** of second node **504**. A Samba URL **808a** identifies a protocol and a location for accessing the "Music" folder. Mount service component **408** in execution environment **401** of first node **502** may interoperate with a Samba client (not shown) included in and/or associated with file system **427**, which in an aspect may be and/or may be included in the data store of execution environment **401** of first node **502**. Samba URL **808a** may include user identification and authentication information to the Samba client if needed. Alternatively, the Samba client and/or mount service **408** may receive such information from the user and/or from a pre-stored profile. In an aspect, the "Music" folder may be mounted as a drive in execution environment **401** of first node **502**.

[0113] In another aspect, a mount descriptor may identify a representation of a data object that is to be removed and/or otherwise unmounted. A first execution environment may detect a mount descriptor that identifies data object in a data store in a second execution environment. It may be determined that the data object is represented at a location in data store of the first execution environment. The representation may be removed based on the mount descriptor.

[0114] In yet another aspect, a mount descriptor may identify a printer accessible to second execution environment. A first execution environment may create a representation of the printer, accessible in the first execution environment, based on the mount descriptor. Accessing the representation of the printer may include accessing the

Motion Offense Exhibit 2004

printer via the network. The printer may be accessed directly and/or indirectly, for example via the second execution environment. Samba as well as other services allows access to printers. Thus a mount descriptor may identify a printer in and/or accessible to execution environment **401** of second node **504** to mount in execution environment **401** of first node **502**.

[0115] As an alternative and/or in case execution environment **401** of first node **502** does not include a Samba client and/or a compatible analog, a mount descriptor may specify one or more alternatives in addition to or instead Samba. For example execution environments may be separated and/or otherwise blocked by one or more firewalls, Mount Service **408** may operate to mount the "Music" found via another identifier in mount descriptor **804a**. Web Distributed Authoring and Versioning (WebDAV) is an extension of the Hypertext Transfer Protocol (HTTP) that facilitates collaboration between users in editing and managing documents and files stored on World Wide Web servers. A WebDAV URL **810a** identifies HTTP as an access protocol and identifies a location in execution environment **401** of second node **504** for accessing the "Music" folder. Mount service component **408** in execution environment **401** of first node **502** may interoperate with a WebDAV client (not shown) included in and/or associated with file system **427**, which in an aspect may be and/or may be included in the data store of execution environment **401** of first node **502**. WebDAV URL **810a** may include user identification and authentication information to the WebDAV client if needed. Alternatively, the WebDAV client and/or mount service **408** may receive such information from the user and/or from a pre-stored profile. In an aspect, the "Music" folder may be mounted as a folder in, for example, a pre-existing Music folder in execution environment **401** of first node **502**. The two music folders may be presented as a merged entity. In another aspect, the "Music" folder identified in the mount descriptor may be mounted at a location in the pre-existing Music folder. The mounted folder may be renamed to distinguish it from a folder with the same name and/or to identify its source by user and/or by node, for example. Alternatives to WebDAV include the file transfer protocol FTP.

Motion Offense Exhibit 2004

[0116] In yet another aspect, a representation of a data object a may include and/or may otherwise be based on a replica of at least a portion of the data object. The replica may be synchronized with other replicas and/or an original data object. A mount service **408** may interoperate with an XtreeemFS client to create a representation of a data object identified in a mount descriptor at location in a data store. XtreemFS replicates file data across multiple storage servers, which can be distributed worldwide. The replication algorithm is designed to handle problems and failure scenarios that can occur in distributed systems including message loss, network partitions, server crashes, and the like. When a replicated file is opened, XtreemFS automatically elects a primary replica for that file which processes all file operations. If the primary fails, one of the backup replicas will automatically take over after a short failover period. To an application accessing a file via XtreemFS a regular file and/or folder and a replicated representation behave the same or similarly.

[0117] For read-only access, replicas of an original folder and/or file may be created and distributed to various replication servers to enhance performance. Through replication a caching infrastructure may be built in order to reduce latency and bandwidth consumption. Partial replicas may be supported, in addition to or instead of full replicas. File, folder, and/or other data object metadata may also be replicated.

[0118] Thus, as describes above, a representation of a remote data object may be created based on a mount descriptor that includes information for accessing the data object based on at least one of Samba, content management interoperability services (CMIS), Web Distributed Authoring and Versioning (WedDAV), Atom Publishing Protocol (AtomPub), file transfer protocol (FTP), RSS, SSH FTP, and an identified network file system.

[0119] Other distributed file systems that may be accessed to mount a representation of a data object identified in a mount descriptor at a location in a data store. The Andrew File System (AFS), the Distributed File System (DFS), and the Network File System (NFS). Atom Publishing Protocol (AtomPub), file transfer protocol (FTP), RSS, SSH

Motion Offense Exhibit 2004

FTP, and content management interoperability services (CMIS) are other means for mounting a data object identified in a mount descriptor.

[0120] With reference to FIG. **2B**, a block **212** illustrates that the method includes receiving, by a second communications agent that represents a second user, data object information that identifies a data object in a second data store in a second execution environment that includes the second communications agent. Accordingly, a system for sharing a data object in a data store via a communication includes means for receiving, by a second communications agent that represents a second user, data object information that identifies a data object in a second data store in a second execution environment that includes the second communications agent. For example, the arrangement in FIG. **3B**, includes descriptor handler component **304** that is operable for receiving, by a second communications agent that represents a second user, data object information that identifies a data object in a second data store in a second execution environment that includes the second communications agent. FIG. **4** illustrates descriptor handler component **404** as an adaptation and/or analog of descriptor handler component **304** in FIG. **3B**. One or more descriptor handler components **404** operate in an execution environment **401**. The system for sharing a data object in a data store via a communication includes one or more processors and logic encoded in one or more tangible media for execution by the one or more processors that when executed is operable for receiving, by a second communications agent that represents a second user, data object information that identifies a data object in a second data store in a second execution environment that includes the second communications agent. In FIG. **4**, a descriptor handler component **404** is illustrated as a component of communications agent **403**.

[0121] FIG. **7** includes data flow message that illustrates a create mount descriptor (createMD) data flow **710** data flow to create and/or otherwise construct a mount descriptor that identifies a data object in a data store of execution environment **401** of second node **504**. Create mount descriptor data flow **710** in FIG. **7** may, in FIG. **4**, represent providing and/or otherwise identifying access information to descriptor

Motion Offense Exhibit 2004

handler component **404** to process in creating a mount descriptor for a data object identified by the data object information. Create mount descriptor data flow **710** may be internal to execution environment **401** of second node **504** and/or may include interoperation by execution environment **401** of second node **504** via a network with another node.

[0122] FIG. **6D**, illustrates an edit window UI element **602d** that, in an aspect, may be presented by communication agent **403** in execution environment **401** of second node **504** in response to a user input to create a new email. In an aspect, a user of communications agent **403** operating in execution environment **401** of second node **504** may provide input corresponding to attach mount UI element **644d** in edit window UI element **602d**. A corresponding UI element handler component **415** may operate to invoke data store navigator component **406** to present a navigation window UI element **646d** to locate one or more data objects in response to user interaction. Data store navigation component **406** may interoperate with one or more UI element handlers **415**, directly or indirectly via descriptor handler component **404**, in an aspect, to present the navigation window UI element **646d** and its contents.

[0123] Navigation window UI element **646d** is presented to receive data object information in response to user input to identify a data object in a data store of execution environment **401** of second node **504**. A folder content pane UI element **648d** in navigation window UI element **646d** is presented including representations of one or more data objects at a location, identified in a path UI element **650d**. A pointer UI element **652d** illustrates a drag and drop operation of a "Music" folder represented in contents pane UI element **648d**, dropped on edit window UI element **602d**. In response, to detecting the drag and drop operation by one or more UI element handler components **415**, data store navigator component **406** may interoperate with the one or more UI element handle components **415** to provide data object information identifying the data object to descriptor handler component **404**.

[0124] Returning to FIG. **2B**, a block **214** illustrates that the method further includes creating a mount descriptor for accessing the data object from the second data store by

Motion Offense Exhibit 2004

a first data store in a first execution environment including a first communications agent that represents a first user. Accordingly, a system for sharing a data object in a data store via a communication includes means for creating a mount descriptor for accessing the data object from the second data store by a first data store in a first execution environment including a first communications agent that represents a first user. For example, the arrangement in FIG. **3B**, includes descriptor generator component **314** that is operable for creating a mount descriptor for accessing the data object from the second data store by a first data store in a first execution environment including a first communications agent that represents a first user. FIG. **4** illustrates descriptor generator component **414** as an adaptation and/or analog of descriptor generator component **314** in FIG. **3B**. One or more descriptor generator components **414** operate in an execution environment **401**. The system for sharing a data object in a data store via a communication includes one or more processors and logic encoded in one or more tangible media for execution by the one or more processors that when executed is operable for creating a mount descriptor for accessing the data object from the second data store by a first data store in a first execution environment including a first communications agent that represents a first user. In FIG. **4**, descriptor generator component **414** is illustrated as a component of communications agent **403**.

[0125] As described above, FIG. **7** illustrates a create mount descriptor (createMD) data flow **710**. Create mount descriptor data flow **710** may result from a request from a remote node, described below, and/or may result in access information received in response to user input, also described above. Access information received by descriptor handler component **404** may be provided to and otherwise identified to a descriptor generator component **414** which may also be included in create mount descriptor data flow **710**. Descriptor generator component **414** operates to create and/or otherwise construct a mount descriptor.

[0126] Data objects located and/or otherwise identified by data object information received by descriptor handler component **404** may be identified to descriptor generator component **414** to construct and/or otherwise create a mount descriptor identifying the

Motion Offense Exhibit 2004

data objects. A descriptor handler component may be a type of content handler component that operates to process data object information to create a mount descriptor that conforms to a schema for creating and/or otherwise constructing a valid mount descriptor. The descriptor generator component **414** may construct and/or otherwise create a mount descriptor based on a schema that defines and/or otherwise identifies a valid mount descriptor for a particular type of communication supported by a communications agent. The mount descriptor is constructed and/or otherwise created, based on the data object information, by the descriptor generator component **414**.

[0127] Descriptor generator component **414** in execution environment **401** of second node **504** may provide the mount descriptor to content manager component **402** to include and/or otherwise identify the mount descriptor in a communication with execution environment **401** of first node **502**, such as an email and/or instant message. Content manager component **402** may interoperate with other types of content handler components **413** to create and/or otherwise construct a message that includes the valid mount descriptor.

[0128] A mount descriptor may be included in a message according to a protocol schema that identifies the message as valid according to a communications protocol. In an aspect, the descriptor generator component **414** may interoperate with the content manager component **402** to create a valid mount descriptor according to a specified schema for the message, so that the message is valid according to a communications protocol supported by a communications protocol component **407** that operates to send the message. As described above, the content manager component **402** may operate in execution environment **401** of second node **504** along with a descriptor generator component **414** to transform data object information into a mount descriptor to include along with data for other parts of a communication into one or more representations suitable for transmitting in a communication, such as a message, to another node, such as first node **502**. Some or all of the representations transmitted are suitable for processing by the communications agent in execution environment **401** of first node **502**. The content manager component **402** in the execution environment **401** of second

Motion Offense Exhibit 2004

node **504** may package the one or more data representations including a representation of the mount descriptor into a message formatted according to the communications protocol.

[0129] A descriptor generator component **414** and/or a content manager component **402** may operate to construct a mount descriptor in a communication. A content type identifier may be included in a position and/or location that identifies a mount descriptor in a communication. The position or location may be absolute or relative. For example, a schema for a communication may define that a mount descriptor in a communication is included in the communication at the end of the communication. There may be one or more mount descriptors at the end. In another aspect, a schema for a communication may specify that a portion of a communication following a particular type of message portion is a mount descriptor. Other mount descriptors may follow. If no mount descriptor is included, the mount descriptor portion may include no content or may include an indicator that no mount descriptor is included.

[0130] FIG. **8A** includes an exemplary content portion **800a** of an email message. Content portion **800a** includes a user message portion **802a** and a mount descriptor portion **804a** described in more detail above. FIG. **8A** illustrates an "application/mount-descriptor" MIME type identifier that may be defined to identify a schema for content including one or more mount descriptors. FIG. **8A** illustrates a mount descriptor **804a** including two identifiers for the "Music" folder. A first identifier **808a** illustrates a Samba URL for accessing the "Music" folder via a Samba file sharing system. The second identifier **810a** illustrates a WebDAV HTTP URL that may allow a receiving node to access the identified data object via a WebDAV server operating in and/or available to an execution environment sending the mount descriptor **804a**. An FTP URL **812a** illustrates a file transfer protocol (FTP) that may allow a receiving node to access the identified data object via a FTP server operating in and/or available to an execution environment sending the mount descriptor **804a**

Motion Offense Exhibit 2004

[0131] Other means are described about with respect to operation of the method illustrated in FIG. **2A**. All such means are suitable for performing the method illustrated in FIG. **2B**.

[0132] Returning to FIG. **2B**, a block **216** illustrates that the method yet further includes including, by the second communications agent, the mount descriptor in a first message, addressed to the first user. Accordingly, a system for sharing a data object in a data store via a communication includes means for including, by the second communications agent, the mount descriptor in a first message, addressed to the first user. For example, the arrangement in FIG. **3B**, includes content manager component **302** that is operable for including, by the second communications agent, the mount descriptor in a first message, addressed to the first user. FIG. **4** illustrates content manager component **402** as an adaptation and/or analog of content manager component **302** in FIG. **3B**. One or more content manager components **402** operate in an execution environment **401**. The system for content manager includes one or more processors and logic encoded in one or more tangible media for execution bye the one or more processors that when executed is operable for including, by the second communications agent, the mount descriptor in a first message, addressed to the first user. In FIG. **4**, a content manager component **402** is illustrated as a component of application **403**.

[0133] As described above, FIG. **7** illustrates first message **702**, including a mount descriptor, sent via network **506** by execution environment **401** of second node **504** to execution environment **401** of first node **502**. In an aspect described below, first message **702** may be sent in response to previous message **708**.

[0134] A mount descriptor generated by a descriptor generator component **414**, along with a user message and any other data to include in a communication, may be provided and/or otherwise identified to content manager component **402** for sending in the communication. The content manager component **402** in execution environment **401** of second node **504** may package data from descriptor generator component **414**

Motion Offense Exhibit 2004

including a representation of the mount descriptor into the message formatted according to the communications protocol.

[0135] Returning to FIG. **2B**, a block **218** illustrates that the method yet further includes sending, via a communications protocol via the network by the second communications agent, the first message for delivery to the first communications agent to create, based on the mount descriptor, a representation, of the data object in the first data storage, that when accessed includes accessing the data object from the second data store. Accordingly, a system for sharing a data object in a data store via a communication includes means for sending, via a communications protocol via the network by the second communications agent, the first message for delivery to the first communications agent to create, based on the mount descriptor, a representation, of the data object in the first data storage, that when accessed includes accessing the data object from the second data store. For example, the arrangement in FIG. **3B**, includes com-out component **318** that is operable for sending, via a communications protocol via the network by the second communications agent, the first message for delivery to the first communications agent to create, based on the mount descriptor, a representation, of the data object in the first data storage, that when accessed includes accessing the data object from the second data store. FIG. **4** illustrates com-out component **418** as an adaptation and/or analog of com-out component **318** in FIG. **3B**. One or more com-out components **418** operate in an execution environment **401**. The system for sharing a data object in a data store via a communication includes one or more processors and logic encoded in one or more tangible media for execution by the one or more processors that when executed is operable for sending, via a communications protocol via the network by the second communications agent, the first message for delivery to the first communications agent to create, based on the mount descriptor, a representation, of the data object in the first data storage, that when accessed includes accessing the data object from the second data store. In FIG. **4**, a com-out component **418** is illustrated as a component of application **403**.

Motion Offense Exhibit 2004

[0136] Com-out component **418** may provide the mount descriptor, the user message, and any other data for sending in the communication in representations suitable for sending by communications protocol component **407** to the communications agent in execution environment **401** of first node **502**. Communications protocol component **407** operating in execution environment **401** of second node **504** may send the data as and/or in a communication, such as an email message, according to the specification(s) of the communications protocol. Communications protocol component **407** may further package and/or otherwise transform the data to send via network stack **405** for delivery via network **506** to execution environment **401** of first node **502**.

[0137] Content manager component **402** may alternatively or additionally encode and/or otherwise transform one or more of the data representations for sending in a data stream such as voice stream and/or a video stream for communicating in a communication with the communications agent in execution environment **401** of first node **502**

[0138] The methods illustrated in FIG. **2A-B** may include additional aspects supported by various adaptations and/or analogs of the arrangement of components in FIG. **3A-B**.

[0139] In an aspect, a mount descriptor may be received as a response to a data object request in a message previously sent. A data object request may include some or all of a keyword expression, a regular expression, an XQuery expression, an XPath expression, a file system path expression, and a structured query language statement.

[0140] The previously sent message may include the data object request in the message separate from a user readable message. The user readable message may be detected by a receiving execution environment along with the data object request. The data object request may be included in the previously sent message according to a schema that identifies the message as valid according a communications protocol by which the message was sent in the communication.

[0141] The data object request may include match information identifying a matching criterion for locating and/or otherwise identifying a data object to be mounted. With respect to FIG. **4** and FIG. **5**, execution environment **401** of first node **502** may request

Motion Offense Exhibit 2004

a mount descriptor from execution environment **401** of second node **504**. FIG. **7** includes a data flow message that illustrates a match information data flow **712** that may occur at least partially in execution environment **401** of first node **502** and/or may include a message (not shown) received via network **506** by execution environment **401** of first node **502**. Match information data flow **712** may represent, in FIG. **4**, a data flow to an identify mount constructor (IMC) component **410** operating in execution environment **401** of first node **502**. The data flow may be via descriptor handler component **404**.

[0142] Sending a message including a data object request may include receiving, by a first execution environment, matching information, identifying a data object matching criterion, from a first user. The message, including a data object request based on the data object matching criterion, may be sent, via the network, to a second execution environment. The message is addressed to the second user. The data object request may be included in the message according to a schema that identifies the previous message as valid according a communications protocol in the communication. A mount descriptor may be received in a subsequent message, in response to sending the data object request.

[0143] In various aspects, IMC component **410** may receive matching information from a user via one or more UI element handler components **415**. The matching information may be received, in response to a user input detected by an input device of an execution environment **401** that includes the IMC component **410**. Alternatively or additionally, matching information may be received by an IMC component in a message received via a network. For example, a user may identify matching information to a browser operating in a client node. The browser may send the matching information via a network communicatively coupling the client node to a web service provider node. The web service provider node may include an IMC component that may operate to receive the matching information identified in the message from the browser. In another aspect, an IMC component may be provided in whole or in part by execution environment **401** of node **502** to a browser in another execution environment.

Motion Offense Exhibit 2004

[0144] FIG. **6E** illustrates an exemplary user interface that may be presented by communications agent **403** operating in execution environment **401** of first node **502**. A UI element handler component **415** may operate to present a user interface element to allow a user to identify matching information. A matching information (MI) UI element **656e** is illustrated in FIG. **6E** as an example. The same or different UI element handler component **415** may present various other UI elements included in MI UI element **656e**. MI UI element **656e** and one or more UI elements it includes may be presented by one or more corresponding UI element handler component(s) **415** that may operate based on a schema that defines valid matching information and/or data object requests for including in a communication via a communications agent **403**. The schema defines one or more rules and/or a vocabulary that defines whether matching information and/or a data object request is valid. Those skilled in the art will understand that numerous languages and/or schemas for providing matching information and/or data object requests currently exist including relatively simple key word based queries to relatively more complicated regular expression languages and data base query languages, such as various versions of structured query language (SQL). New schemas are and will be created, and are, thus, considered within the scope of the subject matter described herein.

[0145] MI UI element **656e** illustrates a user interface for a keyword based schema. Match textbox UI element **658e** illustrates a textbox allowing a user to enter a keyword expression. Various checkbox UI elements **660e** allow a user to define a scope of a data object request by identifying whether a resulting data object request is to be applied to folders, files, and/or content of files in a data store. A communications agent **403** may support more than one data object request schema and/or corresponding matching information schema. A communications agent may provide a user interface to receive valid matching information for each respective schema.

[0146] FIG. **7** includes data flow message that illustrates a create data object request (CIReq) data flow **714** that may be internal to execution environment **401** of first node **502** and/or may include a message received via network **506** by first node **502**. With

Motion Offense Exhibit 2004

respect to FIG. **4**, CIReq data flow **714** may illustrate a data flow to an IMC component **410** operating in execution environment **401** of first node **502**. Match information may be identified to and/or otherwise provided to IMC component **410**.

[0147] An IMC component **414** may be a type of content handler component and/or may be included in a content handler component that operates to process match information to create a data object request that conforms to a schema for creating and/or otherwise constructing a valid data object request. IMC component **410** may construct and/or otherwise create a data object request based on a schema that defines and/or otherwise identifies a valid data object request and/or a valid mount descriptor for a particular type of communication supported by a communications agent. The data object request descriptor is constructed and/or otherwise created, based on the match information, by IMC component **410**. IMC component **410** in execution environment **401** of first node **502** may provide the data object request to content manager component **402** to include and/or otherwise identify the data object request in a communication with execution environment **401** of second node **504**, such as email and/or instant message. Content manager component **402** may interoperate with other types of content handler components **413** to create and/or otherwise construct a message that includes the valid data object request.

[0148] In an aspect, IMC component **410** may interoperate with the content manager component **402** to create the valid data object request according to a specified schema for the message, so that the message is valid according to a communications protocol supported by a communications protocol component **407** that operates to send the message. As described above, the content manager component **402** may operate in execution environment **401** of first node **502** along with IMC component **410** to transform match information into a data object request to include along with data for other parts of a communication into one or more representations suitable for transmitting in a communication, such as a message, to another node, such as second node **504**. Some or all of the representations transmitted are suitable for processing by the communications agent in execution environment **401** of second node **504**. The

Motion Offense Exhibit 2004

content manager component **402** in the execution environment **401** of first node **502** may package the one or more data representations including a representation of the data object request into a message formatted according to the communications protocol.

[0149] In FIG. **8B**, a portion of an email communication **800b** is illustrated formatted as a multipart/mixed content type including data object request portion **802b**. A data object request portion of a communication may be identified as a data object request by its location in the communication and/or by an identifier or markup element, such as a MIME type identifier. A data object request may be detected based on content included in the message and/or based on metadata such as content-type header **804b** identifying a MIME type identifier, such as "application/id-request", which may be defined for representing one or more matching criteria in a data object request. The "application/id-request" MIME type identifier is exemplary. Other MIME type identifiers exist and/or may be defined to identify a data object request in a communication.

[0150] An IMC component and/or a content manager component may operate to construct a data object request in a communication based on XQuery, regular expression, and/or SQL content, to name a few examples. A content type identifier may be included in a position and/or location that identifies a data object request in a communication. The position or location may be absolute or relative. For example, a schema for a communication may define that a data object request in a communication is included in the communication at the end of the communication. There may be one or more data object requests at the end. In another aspect, a schema for a communication may specify that a portion of a communication following a particular type of message portion is a data object request. Other data object requests may follow. If no data object request is included, the data object request portion may include no content or may include an indicator that no data object request is included.

[0151] FIG. **8B** illustrates an "application/id-request" MIME type identifier that may be defined to identify a schema for an XML-based language for specifying id-request XML documents. FIG. **8B** illustrates id-request document **806b**. Id-request document **806b**,

Motion Offense Exhibit 2004

as illustrated, includes criterion tag elements **808b** corresponding to the form elements in a user interface FIG. **6A**. A criterion tag element **808b** identifies a matching criterion name, such as "query" indicating that the criterion is a query expression or a portion thereof. A "type" attribute identifies a schema with the identifier "keyword" for the expression. The criterion tag element **808b** identifies a value for a query "*music*" in a match-expression attribute. Another criterion tag element **808b** specifies a scope query specified in the first criterion tag element. FIG. **8B** illustrates an "and" tag **810b** indicating that all the matching criteria must be met for identifying a data object. An "or" tag (not shown) may be defined by a schema for id-request documents. Other operator elements and operator precedence may be defined by the schema. Grouping elements for managing operator precedence, such as parenthesis element, may be defined by the schema.

[0152] FIG. **7** includes a data flow that illustrates a previous message **716** (with respect to first message **702**) including a data object request. Previous message **716**, as shown, may be sent via network **506** by execution environment **401** of first node **502** to execution environment **401** of second node **504**. In an aspect, first message **702** may be sent in response to previous message **716**.

[0153] In various aspects, a data object request may be based on a date, a time, a length of time, a file type, a database record key, content of the data object, a content type identifier, a format rule, a vocabulary, a role of a user, a security attribute, a location in a data store such as a file system, an attribute of an identified data object, a size, a task, a transaction, a state, a user, a group, a requester, a relationship including a requesting user and a responding user, a keyword, a tag, a folder, and/or a path portion of a data object identifier, to name a few examples.

[0154] A descriptor handler component in an execution environment may receive data object in formation in response to a request from a communications agent in another execution environment. As described above, FIG. **7** illustrates previous message **716**, including a data object request, received via network **506** by execution environment **401** of second node **504** from execution environment **401** of first node **502.**

Motion Offense Exhibit 2004

[0155] In addition to packaging representations of data in a structure and/or format for sending a message according to a communications protocol, a content manager component **402** may operate to detect a data object request received and/or otherwise identified in a message received in a communication. Content manager component **402** operating in execution environment **401** of second node **504** may receive the message, illustrated by previous message **716**, in a communication with execution environment **401** of first node **502**. Com-in component **411** in execution environment **401** of second node **504** may receive the message via communication protocol component **407** and network stack **405**. Previous message **716** may be delivered to execution environment **401** of second node **504** via network **506** based on a communications address of a communicant represented by communications agent **403** in execution environment **401** of second node **504**. A communications agent may represent more than one communicant identified by different addresses.

[0156] Previous message **716** in Fig. **7** may be received in one or more packets via network **506** by network stack **405** and communications protocol component **407** in an instance of and/or analog of execution environment **401** including and/or otherwise provided by second node **504**. The data in the communication may be received by com-in component **411**. Com-in component **411** may provide the data to content manager component **402**. Content manager component **402** may determine one or more content types of the data. The content and/or portions of the content may be provided to one or more content type handler components **413** based on the one or more content types identified by content manager component **402**. For example, FIG. **8B** illustrates message portion **812b** including "text/plain" MIME type identifier **814b** as a content type identifier. Message portion **812b** may be provided to a text/plain content handler **413**. Audio data in a voice communication may be provided to an audio content handler component **413**, and video data in a video communication may be provided to a video content handler component **413**.

[0157] FIG. **7** includes a data flow message that illustrates a detect data object request (detectOReq) data flow **718** that may be internal to execution environment **401** of

Motion Offense Exhibit 2004

second node **504** and/or may include interoperation with another node via a network. With respect to FIG. **4**, detect data object request data flow **718** may correspond to a data exchange and/or other interoperation between content handler component **402** and descriptor handler component **404** to detect and/or otherwise process the data object request received in previous message **716** by execution environment **401** of first node **504**. FIG. **7** further includes a data flow message that illustrates a locate message **720**, that may be included in an aspect of the method illustrated in FIG. **2B**, for locating one or more data objects based on the data object request. The data object request may include and/or otherwise identify a matching criterion. The matching criterion may be identified based on match information identified by the communicant represented by execution environment **401** of first node **502**. The one or more data objects may be located and/or otherwise identified by determining a data object or data objects that match the matching criterion. The data object(s) may be in a data store in and/or otherwise accessible to execution environment **401** of second node **504**, such as one or more files in a file system of execution environment **401** of second node **504**. Locate message **720** may be internal to execution environment **401** of second node **504** and/or may include a message exchanged with a remote device, such as database server and/or a network attached storage device providing storage locations for some or all of a file system of execution environment **401** of second node **504**.

[0158] Com-in component **411** in execution environment **401** of second node **504**, as described above, provides the previous message **716**, received from execution environment **401** of first node **502**, to content manager component **402**. Content manager component **402** is operatively coupled, via com-in component **411**, to communications protocol component **407** for receiving data in communications with other nodes, such as execution environment **401** of first node **502**. Thus, execution environment **401** of second node **504** may receive a message via a com-in component **411** in a communication with execution environment **401** of first node **502**. The message may include a data object request based on match information identified by a user to communications agent **403** in execution environment **401** of first node **502**.

Motion Offense Exhibit 2004

[0159] Data received in a communication, such as in a message, may include data of various content types. In Fig. **4**, communications agent **403** in execution environment **401** of second node **504** includes one or more content handler components **413** to process data received according to its content type. Content manager component **402** may detect and/or otherwise identify a portion of the message as the data object request sent from execution environment **401** of first node **502**. The data object request may be identified by content manager component **402** according to a schema for the message and/or a schema for the data object request. For example, a portion of the message may include a MIME-type identifier that identifies the portion as a data object request and/or otherwise identifies the portion for routing to descriptor handler component **404** operating in and/or otherwise on behalf of execution environment **401** of second node **504**. Alternatively or additionally, the data object request may have structure and/or content that is valid and identifiable according to a schema for defining and/or otherwise identifying valid data object requests. A descriptor handler component may be provided as a type of content handler component, in an aspect. Alternatively or additionally, a descriptor handler component may interoperate with one or more content handler components to detect a data object request in a message received by a content manager component.

[0160] As such, descriptor handler component **404** may operate along with other content handler components **413** to process data representations received in the message **716** from execution environment **401** of first node **502**. Descriptor handler component **404** may also provide transformed data from the representations to one or more user interface element handler components **415**. Descriptor handler component **404** operates to validate and process data object requests according to a schema defining valid data object requests. Various portions of a message including a data object request may be provided to one or more content handler components **413**, including descriptor handler component **404**, to interoperate with presentation controller component **417** in execution environment **401** of second node **504** to present some or all of the received message, including the data object request.

Motion Offense Exhibit 2004

[0161] FIG. **6F** illustrates a window, view window UI element **602f**, presenting an exemplary representation of a message, such as previous message **716**, received in a communication from execution environment **401** of first node **502** by second node **504**. FIG. **6F**, view window UI element **602f** includes a contactor UI element **604f**, a contactee UI element **606f**, and a presentation space **608f** of the view window UI element **602g** that includes a user message UI element **610f** including a text portion of the message provider by the communicant represented by execution environment **401** of first node **502**. The various UI elements may be presented by one or more UI element handler components **415** in communications agent **403** operating in execution environment **401** of second node **504**. A reply UI element **662f** and a reply all UI element **664f** are illustrated as exemplary UI elements in presentation space **608f** presented to allow a user of execution environment **401** of second node **504** to respond and/or otherwise perform an operation based on the received message. FIG. **6F** illustrates a data object request UI interface element **666f** where the data object request identified in the message from execution environment **401** of first node **502** is represented to the communicant represented by the execution environment **401** of second node **504**. In an aspect, the communicant is allowed to authorize the request by providing an input corresponding to a find and reply UI element **668f**. In another aspect, the communicant may allowed to modify the data object request.

[0162] As described herein in an aspect, content manager component **402** may detect content type information to detect a data object request in a communication. For example, the message portion illustrated in FIG. **8B** may be received in previous message **716**. Content manager component **402** may detect "application/id-request" MIME type identifier in content-type header **804b**. The "application/id-request" MIME type identifier may be defined to identify a data object request based on one or more matching criteria for identifying a requested data object. Content manager component **402** may identify data object request portion **802b** as including the data object request.

[0163] In response to detecting a data object request in previous message **716**, content manager component **402** may provide data some or all of the data object

Motion Offense Exhibit 2004

request to descriptor handler component **404**. For example, descriptor handler component **404** may be configured to operate according to a schema defining a format and/or a vocabulary for an XML-based language for id-request documents. Content manager component **402** may provide id-request document **806b**, as a data object request, to descriptor handler component **404**. Descriptor handler component **404** may operate according to the id-request schema. In an aspect, descriptor handler component **404** may process more than one data object request content type. Alternatively or additionally, execution environment **401** may include multiple descriptor handler components **404** for supporting multiple data object request content types.

[0164] As described above, in an aspect, a user may be allowed to select which data objects, if any, are to be identified in a mount descriptor from one or more data objects located and/or otherwise identified by descriptor handler component **404** based on a received data object request. In another aspect, communications agent **403** in Fig. **4** may invoke descriptor handler component **404** automatically to locate one or more data objects based on a matching criterion identified in a data object request and in response to detecting a data object request in a communication, such previous message **716** in FIG. **7**. In another aspect, communications agent **403** may invoke descriptor handler component **404** automatically to locate one or more data objects based on a data object request in a communication in response to detect an indication to present, via a output device, some or all of a user message received in a message in the communication. Descriptor handler component **404** may be invoked and/or may operate based on the data object request and/or matching criterion identified to locate a data object prior to, during, and/or after other content received in the communication is processed. Results of a locate operation may be received as data object information.

[0165] FIG. **6G** illustrates a results window UI element **670g** includes results pane UI element **672g** including representations of two data objects located and/or otherwise identified by data store navigator component **406** based on the data object request. A user may select one or more data objects to identify in a mount descriptor. An attach UI element **674g** may be presented, as illustrated, to allow a user to instruct

Motion Offense Exhibit 2004

communications agent **403** to identify one or more selected data objects in results pane UI element **672g** in a mount descriptor. An attach all UI element **6774g** is illustrated to allow for user input instructing communication agent **403** to identify all data objects represented in the UI element **670g** in one or more mount descriptors in a communication with execution environment **401** of first node **502**. A cancel UI element **678g** allows a user to end presentation of results window UI element **670g**. A user may opt to identify no data objects represented in the dialog UI element **670g** and located and/or otherwise identified by data store navigator component **406** based on the data object request.

[0166] Data objects located and/or otherwise identified by data store navigator component **406** may be identified in data object information to descriptor generator component **414** to construct and/or otherwise create one or more mount descriptors identifying the data objects. As described above, a user may be allowed to select which data objects, if any, are to be identified in a mount descriptor from one or more data objects located and/or otherwise identified by data store navigator component **406**. In another aspect, communications agent **403** in Fig. **4** may invoke data store navigator component **406** automatically to locate one or more data objects based on a matching criterion identified in a data object request and in response to detecting a data object request in a communication. In another aspect, communications agent **403** may invoke data store navigator component **406** automatically to locate one or more data objects based on a data object request in a communication in response to detecting an indication to present, via a output device, some or all of a user readable message received in a message in the communication. Data store navigator component **406** may be invoked and/or may operate based on the data object request and/or matching criterion identified to locate a data object prior to, during, and/or after other content received in the communication is processed.

[0167] In another aspect, a data object request UI element may be presented to receive one or more additional matching criteria from a user. Based on the data object request UI element and input from the user, descriptor handler component **404** may

Motion Offense Exhibit 2004

receive one or more additional matching criterion information and/or may receive input to modify a matching criterion identified in a received data object request. Descriptor handler component **404** may locate one or more data objects in response to receiving the additional criterion.

[0168] In another aspect, a data object located and/or otherwise identified by data object information in response to receiving a data object request must match a query identified by the data object request. Alternatively or additionally, a data object request may include an instruction and/or input for generating a data object, and locating the data object may include generating the data object. Generating a data object may include creating the data object and/or may include modifying and/or otherwise transforming an existing data object. For example, execution environment **401** of second node **504** may include a template stored in a file system. A data object request may identify the template. A descriptor handler component **404** may operate to identify a document or other data object that may be created, is being created, and/or has been created based on the template. In response to a data object request, execution environment **401** of second node **504** may return a mount descriptor that identifies the created and/or modified data object.

[0169] As described above, in various aspects, a data object request may be detected and/or represented based on various syntaxes, grammars, vocabularies, and/or languages. For example, a data object request may be identified and/or represented according to a file system search syntax, a regular expression language, a structured query language (SQL) query, a universal data object identifier schema, an XPATH based language, an XQuery based language, an XML based language, an HTML based language (form-based), and/or a keyword-value pair based language.

[0170] A data object request and/or a mount descriptor in a communication may be communicated via a network according to a first communications protocol. A user message may be exchanged between communicants in the communication via a second communications protocol. For example, a data object request and/or a mount descriptor may be sent on behalf of a first communicant by execution environment of a

Motion Offense Exhibit 2004

first node including a communications agent representing the first communicant. The data object request and/or the mount descriptor may be included in a communication along with a user readable message sent to the execution environment of a second node. The communicants represented by the execution environment of the first node and the execution environment of the second node may communicate via one or more exchanges of audio exchanged via a voice over IP (VoIP) communications protocol.

[0171] Exemplary data objects that may be requested via a data object request include a file, a program component, a data base record, video data, audio data, markup language, binary data, text data, an output of a service. Requested data objects may be pre-existing, volatile, and/or generated in response to the request.

[0172] As described above, receiving match information may include receiving a message, identifying the match information, via network from a node that received match information from the user. For example, match information may be received from a browser by a web server.

[0173] Match information and/or a data object request may be generated, detected, and/or otherwise processed according to a schema that identifies at least one of a rule and a vocabulary that defines a valid data object request. Similarly, a mount descriptor may be generated, detected, and/or otherwise processed according to a schema. One or more schemas may alone or together define one or more of valid data object information, match information, a data object request, and/or a mount descriptor. A schema, as just described, may be identified based on a data received and/or otherwise identified by an execution environment hosting a communications agent via the network from another execution environment hosting communications agent and/or from a path node included in a network path in the network communicatively coupling the node to one or more nodes that respectively host a communications agent.

[0174] In a further aspect, multiple alternative schemas may be defined for each of data object information, match information, data object requests, and/or mount descriptors. A schema may be selected and/or otherwise identified from the multiple schemas based by execution environment **401** of first node **502** and the execution

Motion Offense Exhibit 2004

environment **401** of second node **504** and/or based on information exchanged by the two nodes. For example, a schema may be selected by the first communicant and/or the second communicant. One may inform the other, via a communications protocol and/or via any other suitable means. The communicants may negotiate which schema to use via any suitable means.

[0175] The communications agents in execution environment **401** of second node **504** and execution environment **401** of first node **502** may select a schema For example, a schema may be identified and/or otherwise selected based on a communications protocol supported and/or a communications protocol not supported by one or both nodes. That is, in a negotiation to select communication protocol to use allowing communicants to exchange information, a schema may be identified. A schema may be identified and/or otherwise selected based on an attribute of the data object. A schema for identifying image data may be different than a schema for editable documents. A schema for identifying a data object in a first file system may be different than a schema for identifying a data object in a different data store.

[0176] As described above. In an aspect match information and/or a data object request, data object information, and/or a mount descriptor may identify a scope that specifies all or some portion of one or more data stores for locating, identifying, and/or otherwise accessing a data object. For example, a communication agent **403** in execution environment **401** of first node **502** may receive user input that identifies a folder in a file system of the execution environment **401** of second node **504** that defines a scope for locating a data object.

[0177] Match information and/or a data object request be represented in a representation that includes a complete or a portion of at least one of a keyword expression, a regular expression, an XQuery expression, an XPath expression, a file system path expression, and a structured query language statement..

[0178] Further as also described above, a data object request and/or a mount descriptor may be included in a message according to a schema that identifies the message as valid for a specified communications protocol. Those skilled in the art are

Motion Offense Exhibit 2004

aware that schemas for email differ from schemas for instant messages and from streamed audio, for example.

[0179] A data object request and/or a mount descriptor may be exchanged in a communication in which audio data and/or image data is also exchanged. Multimedia may be a packaged to include one or more data streams which may include text based data streams. A data object request and/or a mount descriptor may be included in an email, an instant message, a multi-media message, a short-message service message, and/or a data stream. The data stream may include audio data, image data, and/or text data.

[0180] A data object request and/or a mount descriptor may be included in and/or detected in a communication based on a location in the communication and/or based on a marker, such as an identifier in the communication that identifies a location for the data object request, the mount descriptor, an access request, and/or an access response. The location may be absolute, such as fixed number of bytes or fields from the start or end of a message. Alternatively or additionally, the location may be identified by a location relative to another detectable portion of the communication. FIGs. **8A-B** illustrate the user of MIME type identifiers as markers that identify locations for one or more of a data object request and/or a mount descriptor in a communication.

[0181] A data object request and/or a mount descriptor or respective portions thereof may include at least a portion of a keyword expression, a regular expression, expression including a Boolean operator, an expression including a precedence information, and a structured query language statement.

[0182] With respect to the method illustrated in FIG. **2B**, the method as described above may include locating and/or otherwise identifying a data object automatically in response to receiving a data object request that matches and/or otherwise identifies the data object. Further, a mount descriptor may be created and/or otherwise generated automatically, in response to locating and/or otherwise identifying the data object.

[0183] In an aspect, in response to receiving and/or otherwise detecting a data object request, a user interface may be presented based on the data object request to prompt

Motion Offense Exhibit 2004

a user to process the data object request in some manner, such as authorizing the request and/or modify the request by restricting its scope. In response to detecting user input targeting and/or otherwise corresponding to the presented user interface, one or more data object may be located and/or otherwise identified based on the data object request.

[0184] A user may change a received data object request. One or more data object may be located and/or otherwise identified based on the changed data object request.

[0185] In a further aspect, once one or more data objects a located and/or otherwise identified based on a data object request and/or otherwise by data object information, a user interface may be presented to allow a user to identify the one or more data objects. Information from a user may be received that selects one or more data objects to identify in a mount descriptor and/or to not identify in the mount descriptor. A user, in an aspect, may be allowed to identify a data object not located and/or otherwise identified by the user's node. The data object, based on the user input, is identified in the mount descriptor.

[0186] In light of the description provided above, a mount descriptor may be generated automatically in response to detecting a corresponding data object request by an execution environment **401** of first node. Further, the automatically generated mount descriptor may be sent to the node that sent the corresponding data object request,

[0187] The methods illustrated in FIG. **2A-B** may yet further include additional aspects supported by various adaptations and/or analogs of the arrangement of components in FIG. **3A-B**. Performing the method illustrated in FIG. **2A** and/or the method illustrated in FIG. **2B** and/or any of its extension and/or in any of its aspects may include one or more of calling a function or method of an object, sending a message via a network; sending a message via an inter-process communication mechanism such as a pipe, a semaphore, a shared data area, and/or a queue; and/or receiving a request such as poll and responding to invoke, and sending an asynchronous message.

[0188] Output devices suitable for presenting a representation of a data object and/or a tag include a visual output device, an audio output device, and a tactile output device.

Motion Offense Exhibit 2004

One output device may present a data object and another output device may present a tag with which the data object is tagged.

[0189] Any action included in performing the method illustrated in FIG. **2A** as well as any action included in in performing the method illustrated in FIG. **2B** may include sending and/or receiving a message via a network. Further, a message, included in performing any of the subject matter described herein and/or any of its extensions in any of its aspects, may an asynchronous message without a corresponding request.

[0190] To the accomplishment of the foregoing and related ends, the descriptions and annexed drawings set forth certain illustrative aspects and implementations of the disclosure. These are indicative of but a few of the various ways in which one or more aspects of the disclosure may be employed. The other aspects, advantages, and novel features of the disclosure will become apparent from the detailed description included herein when considered in conjunction with the annexed drawings.

[0191] It should be understood that the various components illustrated in the various block diagrams represent logical components that are configured to perform the functionality described herein and may be implemented in software, hardware, or a combination of the two. Moreover, some or all of these logical components may be combined, some may be omitted altogether, and additional components may be added while still achieving the functionality described herein. Thus, the subject matter described herein may be embodied in many different variations, and all such variations are contemplated to be within the scope of what is claimed.

[0192] To facilitate an understanding of the subject matter described above, many aspects are described in terms of sequences of actions that may be performed by elements of a computer system. For example, it will be recognized that the various actions may be performed by specialized circuits or circuitry (e.g., discrete logic gates interconnected to perform a specialized function), by program instructions being executed by one or more processors, or by a combination of both. The description herein of any sequence of actions is not intended to imply that the specific order described for performing that sequence must be followed.

Motion Offense Exhibit 2004

[0193] Moreover, the methods described herein may be embodied in executable instructions stored in a tangible computer readable medium for use by or in connection with an instruction execution machine, system, apparatus, or device, such as a computer-based or processor-containing machine, system, apparatus, or device. As used here, a "tangible computer readable medium" may include one or more of any suitable media for storing the executable instructions of a computer program in one or more forms including an electronic, magnetic, optical, and electromagnetic form, such that the instruction execution machine, system, apparatus, or device may read (or fetch) the instructions from the tangible computer readable medium and execute the instructions for carrying out the described methods. A non-transitory computer readable storage medium is tangible. A non-exhaustive list of conventional exemplary tangible computer readable storage media includes a portable computer diskette; a random access memory (RAM); a read only memory (ROM); an erasable programmable read only memory (EPROM or Flash memory); optical storage devices, including a portable compact disc (CD), a portable digital video disc (DVD), a high definition DVD (HD-DVD.TM.), and a Blu-ray.TM. disc; and the like.

[0194] Thus, the subject matter described herein may be embodied in many different forms, and all such forms are contemplated to be within the scope of what is claimed. It will be understood that various details may be changed without departing from the scope of the claimed subject matter. Furthermore, the foregoing description is for the purpose of illustration only, and not for the purpose of limitation, as the scope of protection sought is defined by the claims as set forth hereinafter together with any equivalents.

[0195] All methods described herein may be performed in any order unless otherwise indicated herein explicitly or by context. The use of the terms "a" and "an" and "the" and similar referents in the context of the foregoing description and in the context of the following claims are to be construed to include the singular and the plural, unless otherwise indicated herein explicitly or clearly contradicted by context. The foregoing

Motion Offense Exhibit 2004

description is not to be interpreted as indicating that any non-claimed element is essential to the practice of the subject matter as claimed.

Motion Offense Exhibit 2004

PTO/SB/08a (01-10)
Approved for use through 07/31/2012. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	
Filing Date	2012-09-25
First Named Inventor	Robert Paul Morris
Art Unit	
Examiner Name	
Attorney Docket Number	DRV0010

U.S.PATENTS Remove

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	7133897	B1	2006-11-07	Tran	
	2	7209953	B2	2007-04-24	Brooks	

If you wish to add additional U.S. Patent citation information please click the Add button. Add

U.S.PATENT APPLICATION PUBLICATIONS Remove

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20080256199	A1	2008-10-16	Pesala	
	2	20100011077	A1	2010-01-14	Shkolnikov	
	3	20050114664	A1	2005-05-26	Davin	
	4	20060256934	A1	2006-11-16	Mazor	

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	2012-09-25
	First Named Inventor	Robert Paul Morris
	Art Unit	
	Examiner Name	
	Attorney Docket Number	DRV0010

	5	20040064478	A1	2004-04-01	Canesi	
	6	20040186894	A1	2004-09-23	Jhingan	
	7	20050010646	A1	2005-01-13	Shiina	
	8	20050102348	A1	2005-05-12	Parsons	
	9	20070078902	A1	2007-04-05	Buschi	
	10	20080201364	A1	2008-08-21	Buschi	
	11	20100070602	A1	2010-05-18	Malik	
	12	20110276637	A1	2011-11-10	Thornton	

If you wish to add additional U.S. Published Application citation information please click the Add button. Add

FOREIGN PATENT DOCUMENTS Remove

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1							☐

If you wish to add additional Foreign Patent Document citation information please click the Add button Add

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	
	Filing Date	2012-09-25
	First Named Inventor	Robert Paul Morris
	Art Unit	
	Examiner Name	
	Attorney Docket Number	DRV0010

NON-PATENT LITERATURE DOCUMENTS			Remove
Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1	REED, N, et al, "Multipurpose Internet Mail Extensions (MIME) Part 1: Format of Internet Message Bodies", RFC 2045, pages 1-31, 1996-11, Internet Engineering Task Force (IEFT), http://www.ietf.org/rfc/rfc2045.txt	☐
	2	RESNICK, E,, "Internet Message Format", RFC 5322, pages 1-57, 2008-10, Internet Engineering Task Force (IEFT), http://www.ietf.org/rfc/rfc5322.txt	☐

If you wish to add additional non-patent literature document citation information please click the Add button Add

EXAMINER SIGNATURE			
Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	
Filing Date	2012-09-25
First Named Inventor	Robert Paul Morris
Art Unit	
Examiner Name	
Attorney Docket Number	DRV0010

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

☐ That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

☐ See attached certification statement.

☐ The fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

☒ A certification statement is not submitted herewith.

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Robert Paul Morris/	Date (YYYY-MM-DD)	2012-09-25
Name/Print	Robert Paul Morris	Registration Number	

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Electronic Patent Application Fee Transmittal

Application Number:	
Filing Date:	
Title of Invention:	Methods, Systems, and Program Products for Sharing a Data Object in a Data Store Via a Communication
First Named Inventor/Applicant Name:	Robert Paul Morris
Filer:	Robert Paul Morris
Attorney Docket Number:	DRV0010

Filed as Small Entity

Utility under 35 USC 111(a) Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Utility filing Fee (Electronic filing)	4011	1	95	95
Utility Search Fee	2111	1	310	310
Utility Examination Fee	2311	1	125	125
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				
Miscellaneous:				
		Total in USD ($)		**530**

Electronic Acknowledgement Receipt

EFS ID:	13836074
Application Number:	13626635
International Application Number:	
Confirmation Number:	2210
Title of Invention:	Methods, Systems, and Program Products for Sharing a Data Object in a Data Store Via a Communication
First Named Inventor/Applicant Name:	Robert Paul Morris
Customer Number:	108424
Filer:	Robert Paul Morris
Filer Authorized By:	
Attorney Docket Number:	DRV0010
Receipt Date:	25-SEP-2012
Filing Date:	
Time Stamp:	18:00:33
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	Credit Card
Payment was successfully received in RAM	$530
RAM confirmation Number	5527
Deposit Account	505943
Authorized User	MORRIS, ROBERT PAUL

The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows:

Charge any Additional Fees required under 37 C.F.R. Section 1.17 (Patent application and reexamination processing fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.19 (Document supply fe

Charge any Additional Fees required under 37 C.F.R. Section 1.20 (Post Issuance fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.21 (Miscellaneous fees and charges)

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Abstract	0264-Abstract.pdf	28093 740d70066b8f1ce6de0a15c29eee7685b629dafd	no	1
Warnings:					
Information:					
2	Application Data Sheet	0264-ADS.pdf	1882013 7e24eb941d0ff69fde63dcf9c8f9ceff68ed47f6	no	6
Warnings:					
Information:					
3	Claims	0264-Claims.pdf	52401 68f2d556476c3a91463ed4998d486aacd4cf1c71	no	7
Warnings:					
Information:					
4	Drawings-only black and white line drawings	0264-Drawings.pdf	121712 ba356b9bfe155f7e287f3d8def8b035dfb479507	no	15
Warnings:					
Information:					
5	Oath or Declaration filed	0264-Oath.pdf	63030 2fa9fa415c009d06bc74329c87b460303a6cbd50	no	2
Warnings:					
Information:					
6	Specification	0264-Specification.pdf	268890 fb23342001976126a385534fdd79a08d00927f40	no	65
Warnings:					
Information:					
7	Information Disclosure Statement (IDS) Form (SB08)	0264-US_IDS_Form__SB_08a.pdf	612996 1ab77c1b881c443428954dd7d42ce1619470c18e	no	5
Warnings:					
Information:					
8	Non Patent Literature	0264-IDS-NPL-RFC2045.pdf	45030 ca4d1c409fb46ef84ba6643fe0b5776c05332975	no	32
Warnings:					

Information:					
9	Non Patent Literature	0264-IDS-NPL-RFC5322.pdf	69075 463baecd6adeb9c9c734ce99588c5cf25eafb0e3	no	58
Warnings:					
Information:					
10	Fee Worksheet (SB06)	fee-info.pdf	32481 4b97665c3c85c1d2e4006fc3940495922b9d60b1	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	3175721		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

ABSTRACT

Methods and systems are described for sharing a data object in a data store via a communication. In an aspect, a second communications agent receives data object information that identifies a data object in a second data store in a second execution environment. A mount descriptor for accessing the data object is created. A first message is sent from the second communications agent and received by a first communications agent in a first execution environment according to a first communication protocol. The mount descriptor is detected and a first location in a first data store in the first execution environment is determined. A representation of the data object is created, based on the mount descriptor, at the location. The data object is accessible to the first execution environment via the representation.

Motion Offense Exhibit 2004

PTO/AIA/14 (08-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	DRV0010
	Application Number	
Title of Invention	Methods, Systems, and Program Products for Sharing a Data Object in a Data Store Via a Communication	

The application data sheet is part of the provisional or nonprovisional application for which it is being submitted. The following form contains the bibliographic data arranged in a format specified by the United States Patent and Trademark Office as outlined in 37 CFR 1.76.
This document may be completed electronically and submitted to the Office in electronic format using the Electronic Filing System (EFS) or the document may be printed and included in a paper filed application.

Secrecy Order 37 CFR 5.2

☐ Portions or all of the application associated with this Application Data Sheet may fall under a Secrecy Order pursuant to 37 CFR 5.2 (Paper filers only. Applications that fall under Secrecy Order may not be filed electronically.)

Inventor Information:

Inventor 1 Remove

Legal Name

Prefix	Given Name	Middle Name	Family Name	Suffix
	Robert	Paul	Morris	

Residence Information (Select One) ◉ US Residency ○ Non US Residency ○ Active US Military Service

City	Raleigh	State/Province	NC	Country of Residence i	US

Mailing Address of Inventor:

Address 1	712 Latta Street		
Address 2			
City	Raleigh	State/Province	NC
Postal Code	27607	Country i	US

All Inventors Must Be Listed - Additional Inventor Information blocks may be generated within this form by selecting the **Add** button. Add

Correspondence Information:

Enter either Customer Number or complete the Correspondence Information section below.
For further information see 37 CFR 1.33(a).

☐ **An Address is being provided for the correspondence Information of this application.**

Customer Number	108424		
Email Address	paul.morris@deeprv.com	Add Email	Remove Email
Email Address	paul.morris@nc.rr.com	Add Email	Remove Email

PTO/AIA/14 (08-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	DRV0010
	Application Number	
Title of Invention	Methods, Systems, and Program Products for Sharing a Data Object in a Data Store Via a Communication	

Application Information:

Title of the Invention	Methods, Systems, and Program Products for Sharing a Data Object in a Data Store Via a Communication		
Attorney Docket Number	DRV0010	**Small Entity Status Claimed**	☒
Application Type	Nonprovisional		
Subject Matter	Utility		
Suggested Class (if any)		**Sub Class (if any)**	
Suggested Technology Center (if any)			
Total Number of Drawing Sheets (if any)	15	**Suggested Figure for Publication (if any)**	

Publication Information:

☐ Request Early Publication (Fee required at time of Request 37 CFR 1.219)

☐ **Request Not to Publish.** I hereby request that the attached application not be published under 35 U.S.C. 122(b) and certify that the invention disclosed in the attached application **has not and will not** be the subject of an application filed in another country, or under a multilateral international agreement, that requires publication at eighteen months after filing.

Representative Information:

Representative information should be provided for all practitioners having a power of attorney in the application. Providing this information in the Application Data Sheet does not constitute a power of attorney in the application (see 37 CFR 1.32).
Either enter Customer Number or complete the Representative Name section below. If both sections are completed the customer Number will be used for thr Representative Information during processing.

Please Select One:	◉ Customer Number	○ US Patent Practitioner	○ Limited Recognition (37 CFR 11.9)
Customer Number	108424		

Domestic Benefit/National Stage Information:

This section allows for the applicant to either claim benefit under 35 U.S.C. 119(e), 120, 121, or 365(c) or indicate National Stage entry from a PCT application. Providing this information in the application data sheet constitutes the specific reference required by 35 U.S.C. 119(e) or 120, and 37 CFR 1.78.

Prior Application Status			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)

Additional Domestic Benefit/National Stage Data may be generated within this form by selecting the **Add** button. Add

Foreign Priority Information:

PTO/AIA/14 (08-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	DRV0010
	Application Number	
Title of Invention	Methods, Systems, and Program Products for Sharing a Data Object in a Data Store Via a Communication	

This section allows for the applicant to claim benefit of foreign priority and to identify any prior foreign application for which priority is not claimed. Providing this information in the application data sheet constitutes the claim for priority as required by 35 U.S.C. 119(b) and 37 CFR 1.55(a).

Remove

Application Number	Country [i]	Filing Date (YYYY-MM-DD)	Priority Claimed
			(•) Yes ( ) No

Additional Foreign Priority Data may be generated within this form by selecting the **Add** button. Add

Authorization to Permit Access:

☐ Authorization to Permit Access to the Instant Application by the Participating Offices

If checked, the undersigned hereby grants the USPTO authority to provide the European Patent Office (EPO), the Japan Patent Office (JPO), the Korean Intellectual Property Office (KIPO), the World Intellectual Property Office (WIPO), and any other intellectual property offices in which a foreign application claiming priority to the instant patent application is filed access to the instant patent application. See 37 CFR 1.14(c) and (h). This box should not be checked if the applicant does not wish the EPO, JPO, KIPO, WIPO, or other intellectual property office in which a foreign application claiming priority to the instant patent application is filed to have access to the instant patent application.

In accordance with 37 CFR 1.14(h)(3), access will be provided to a copy of the instant patent application with respect to: 1) the instant patent application-as-filed; 2) any foreign application to which the instant patent application claims priority under 35 U.S.C. 119(a)-(d) if a copy of the foreign application that satisfies the certified copy requirement of 37 CFR 1.55 has been filed in the instant patent application; and 3) any U.S. application-as-filed from which benefit is sought in the instant patent application.

In accordance with 37 CFR 1.14(c), access may be provided to information concerning the date of filing this Authorization.

Applicant Information:

Providing assignment information in this section does not substitute for compliance with any requirement of part 3 of Title 37 of CFR to have an assignment recorded by the Office.

PTO/AIA/14 (08-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	DRV0010
	Application Number	
Title of Invention	Methods, Systems, and Program Products for Sharing a Data Object in a Data Store Via a Communication	

Applicant 1

If the applicant is the inventor (or the remaining joint inventor or inventors under 37 CFR 1.45), this section should not be completed. The information to be provided in this section is the name and address of the legal representative who is the applicant under 37 CFR 1.43; or the name and address of the assignee, person to whom the inventor is under an obligation to assign the invention, or person who otherwise shows sufficient proprietary interest in the matter who is the applicant under 37 CFR 1.46. If the applicant is an applicant under 37 CFR 1.46 (assignee, person to whom the inventor is obligated to assign, or person who otherwise shows sufficient proprietary interest) together with one or more joint inventors, then the joint inventor or inventors who are also the applicant should be identified in this section.

Remove

(•) Assignee	( ) Legal Representative under 35 U.S.C. 117
( ) Person to whom the inventor is obligated to assign.	( ) Person who shows sufficient proprietary interest

If applicant is the legal representative, indicate the authority to file the patent application, the inventor is:

Name of the Deceased or Legally Incapacitated Inventor :

If the Assignee is an Organization check here. [X]

Organization Name	Deep River Ventures, LLC		
Mailing Address Information:			
Address 1	712 Latta Street		
Address 2			
City	Raleigh	**State/Province**	NC
Country [i]	US	Postal Code	27607
Phone Number	919-348-2042	Fax Number	
Email Address	paul.morris@deeprv.com		

Additional Applicant Data may be generated within this form by selecting the Add button. Add

Signature:

Remove

NOTE: This form must be signed in accordance with 37 CFR 1.33. See 37 CFR 1.4 for signature requirements and certifications

Signature	/Robert Paul Morris/			Date (YYYY-MM-DD)	2012-09-25
First Name	Robert	Last Name	Morris	Registration Number	

Additional Signature may be generated within this form by selecting the Add button. Add

Motion Offense Exhibit 2004

PTO/AIA/14 (08-12)
Approved for use through 01/31/2014. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	DRV0010
	Application Number	
Title of Invention	Methods, Systems, and Program Products for Sharing a Data Object in a Data Store Via a Communication	

This collection of information is required by 37 CFR 1.76. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 23 minutes to complete, including gathering, preparing, and submitting the completed application data sheet form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these records.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

I CLAIM:

1. A method for sharing a data object in a data store via a communication, the method comprising:

receiving, by a first communications agent in a first execution environment according to a first communication protocol via a network from a second communications agent in a second execution environment, a first message, in a communication and addressed to a first user represented by the first communications agent, that includes a mount descriptor that identifies a data object in a second data store in the second execution environment;

detecting the mount descriptor;

determining a first location in a first data store in the first execution environment; and

creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store,

wherein performing at least one of the preceding actions comprising the method includes execution of an instruction by a processor.

2. The method of claim 1 wherein the communication includes an exchange via the network between the first execution environment and the second execution environment of at least one of audio data, image data, and text data input by at least one of the first user and a second user represented by the second communications

agent via at least one of the first communications protocol and a second communication protocol.

3. The method of claim 1 wherein receiving the first message includes:

receiving, by the first execution environment, match information, identifying a data object matching criterion, from the first user;

sending, via the network, a previous message, to the second communications agent, including a data object request, based on the data object matching criterion, wherein the previous message is addressed to the second user; and

receiving the first message in response to sending the previous message.

4. The method of claim 3 wherein the data object request includes information for the second execution environment for at least one of generating and modifying the data object.

5. The method of claim 3 wherein the data object request identifies a scope specifying at least a portion of the second data store.

6. The method of claim 1 wherein the mount descriptor is valid according to schema that identifies at least one of a rule and a vocabulary for at least one of determining that the mount descriptor is valid and for determining an identifier of the data object and the schema is identified based on at least one of the first communications protocol and an attribute of the data object.

Motion Offense Exhibit 2004

7. The method of claim 1 wherein receiving the first message includes: detecting the mount descriptor in message along with a separate user readable message from the second user.

8. The method of claim 1 wherein determining the first location at least one of is determined automatically and includes performing a drag and drop operation on a representation of the data object presented via an output device, wherein the first location is a target of the drop operation.

9. The method of claim 1 wherein creating the representation includes:

receiving an indication by the first communications agent to access the data object;

automatically determining the first location in response to receiving the indication;

automatically creating the representation at the location, in response to receiving the indication; and

accessing the data object via the network based on the mount descriptor.

10. The method of claim 1 further includes;

accessing the representation;

sending a second message to the second execution environment to access the data object; and

receiving at least a portion of the data object in a third message.

Motion Offense Exhibit 2004

11. A method for sharing a data object in a data store via a communication, the method comprising:

receiving, by a second communications agent that represents a second user, data object information that identifies a data object in a second data store in a second execution environment that includes the second communications agent;;

creating a mount descriptor for accessing the data object from the second data store by a first data store in a first execution environment including a first communications agent that represents a first user;

including, by the second communications agent, the mount descriptor in a first message, addressed to the first user; and

sending, via a communications protocol via the network by the second communications agent, the first message for delivery to the first communications agent to create, based on the mount descriptor, a representation, of the data object in the first data storage, that when accessed includes accessing the data object from the second data store,

wherein performing at least one of the preceding actions comprising the method includes execution of an instruction by a processor.

12. The method of claim 11 wherein receiving the data object information includes receiving, via the network, a previous message, by the second communications agent, including a data object request identifying data object information, wherein the previous message is addressed to the second user.

Motion Offense Exhibit 2004

13. The method of claim 12 wherein presenting the locator user interface element includes:

presenting a locator user interface element based on the data object request;

receiving data object request change information in response to a user input, detected via an input device;

changing the data object request; and

locating the data object based on the changed data object request.

14. The method of claim 11 wherein the data object information includes information for the second execution environment for at least one of generating and modifying the data object.

15. The method of claim 11 wherein the data object information identifies a scope specifying at least a portion of the second data store.

16. The method of claim 11 wherein including the mount descriptor in the first message includes including the mount descriptor in the first message separate from a user readable message included in the first message.

17. The method of claim 11 wherein the communication includes an exchange via the network between the first execution environment and the second execution environment of at least one of audio data, image data, and text data input by at least one of the first user and a second user represented by the second communications agent via at least one of the first communications protocol and a second communication protocol.

Motion Offense Exhibit 2004

18. The method of claim 11 further includes:

receiving a second message to the second execution environment to access the data object;

accessing at least a portion of the data object; and

sending the at least a portion of the data object to the first execution environment in a third message.

19. A system for sharing a data object in a data store via a communication, the system comprising:

a content manager component or receiving, by a first communications agent in a first execution environment according to a first communication protocol via a network from a second communications agent in a second execution environment, a first message, in a communication and addressed to a first user represented by the first communications agent, that includes a mount descriptor that identifies a data object in a second data store in the second execution environment;

a descriptor handler component for detecting the mount descriptor;

a data store navigator component for determining a first location in a first data store in the first execution environment;

a mount service component for creating, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store; and

a processor, wherein at least one of the content manager component, the descriptor handler component, the data store navigator component, and the mount

service component includes an instruction that is executed by the processor during operation of the system.

20. A tangible computer readable storage medium embodying a computer program, executable by a machine, for sharing a data object in a data store via a communication, the computer program comprising executable instructions for:

receiving, by a second communications agent that represents a second user, data object information that identifies a data object in a second data store in a second execution environment that includes the second communications agent;

creating a mount descriptor for accessing the data object from the second data store by a first data store in a first execution environment including a first communications agent that represents a first user;

including, by the second communications agent, the mount descriptor in a first message, addressed to the first user; and

sending, via a communications protocol via the network by the second communications agent, the first message for delivery to the first communications agent to create, based on the mount descriptor, a representation, of the data object in the first data storage, that when accessed includes accessing the data object from the second data store

Motion Offense Exhibit 2004

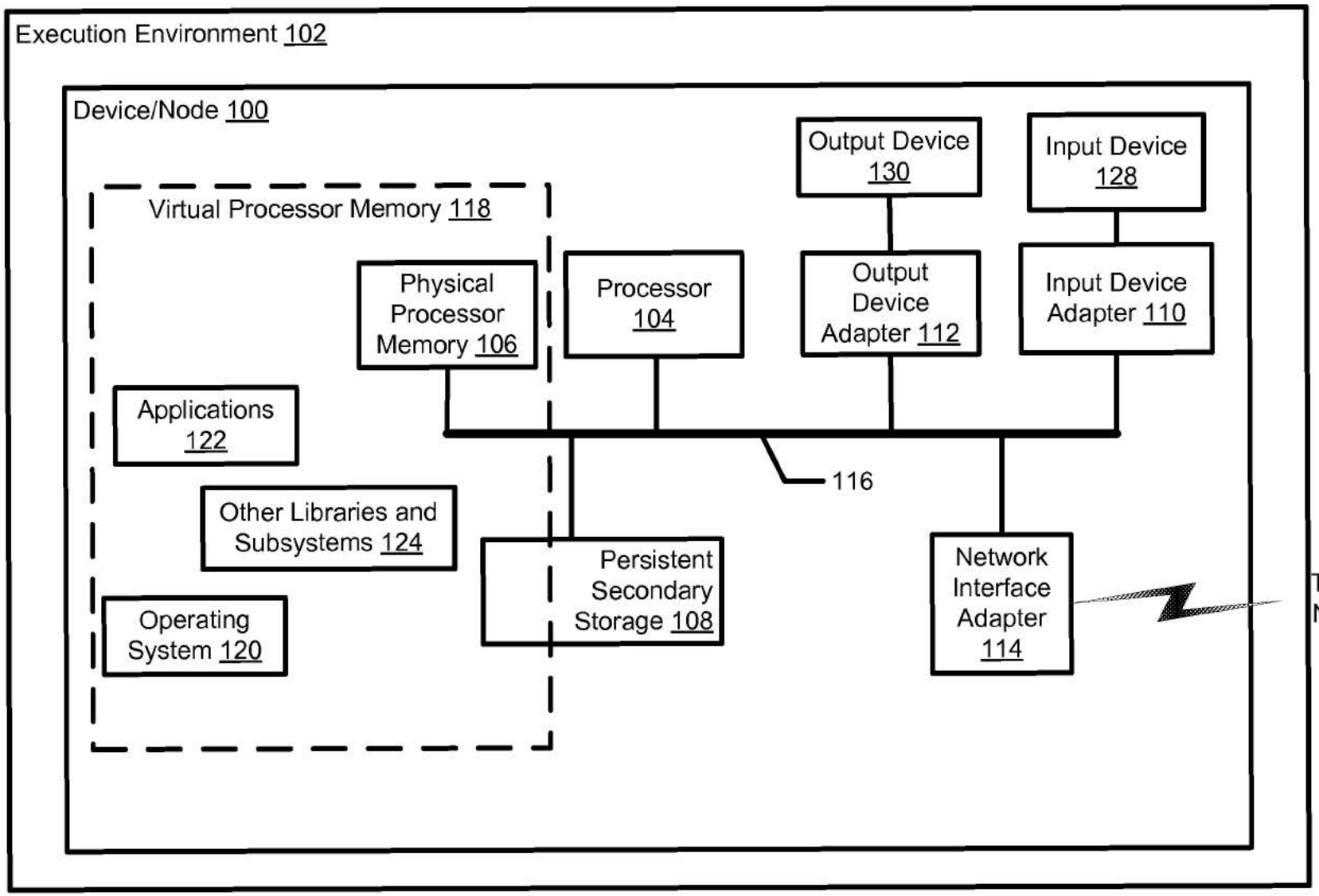

Execution Environment 102
Device/Node 100
Virtual Processor Memory 118
Physical Processor Memory 106
Processor 104
Output Device 130
Input Device 128
Output Device Adapter 112
Input Device Adapter 110
Applications 122
Other Libraries and Subsystems 124
Operating System 120
116
Persistent Secondary Storage 108
Network Interface Adapter 114
To/From Network

FIG. 1

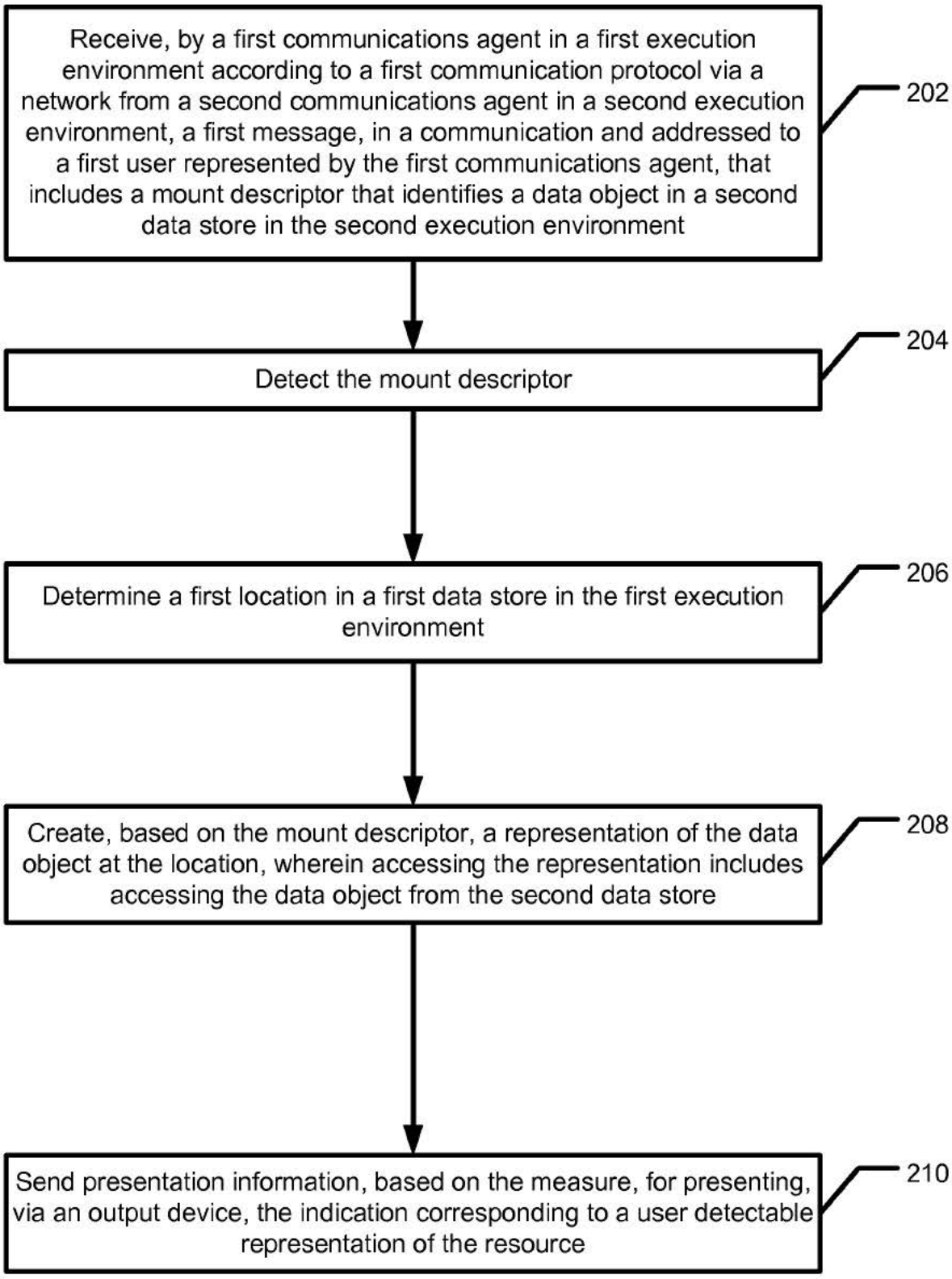

Receive, by a first communications agent in a first execution environment according to a first communication protocol via a network from a second communications agent in a second execution environment, a first message, in a communication and addressed to a first user represented by the first communications agent, that includes a mount descriptor that identifies a data object in a second data store in the second execution environment
202
Detect the mount descriptor
204
Determine a first location in a first data store in the first execution environment
206
Create, based on the mount descriptor, a representation of the data object at the location, wherein accessing the representation includes accessing the data object from the second data store
208
Send presentation information, based on the measure, for presenting, via an output device, the indication corresponding to a user detectable representation of the resource
210

FIG. 2A

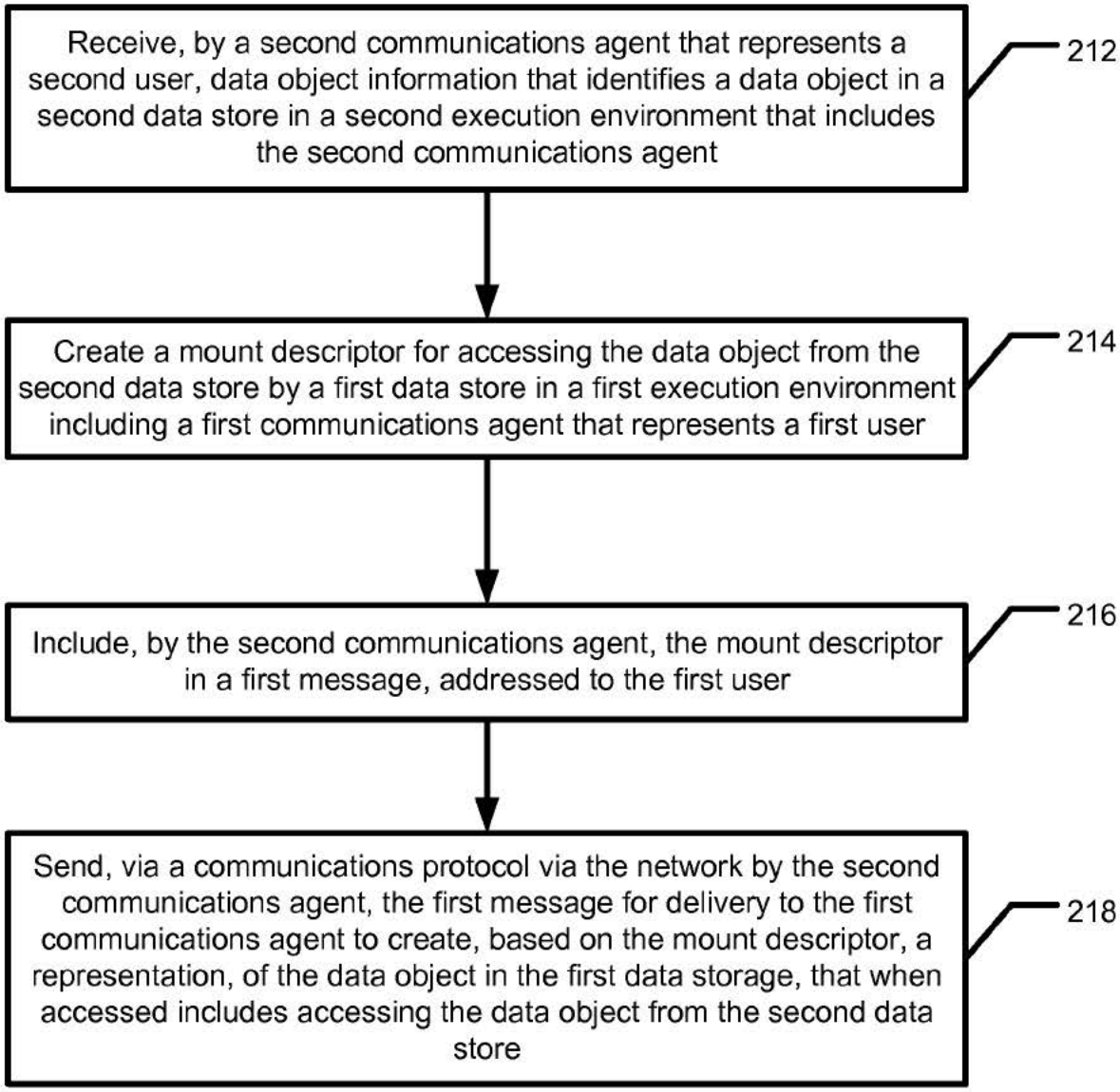

Receive, by a second communications agent that represents a second user, data object information that identifies a data object in a second data store in a second execution environment that includes the second communications agent
212
Create a mount descriptor for accessing the data object from the second data store by a first data store in a first execution environment including a first communications agent that represents a first user
214
Include, by the second communications agent, the mount descriptor in a first message, addressed to the first user
216
Send, via a communications protocol via the network by the second communications agent, the first message for delivery to the first communications agent to create, based on the mount descriptor, a representation, of the data object in the first data storage, that when accessed includes accessing the data object from the second data store
218

FIG. 2B

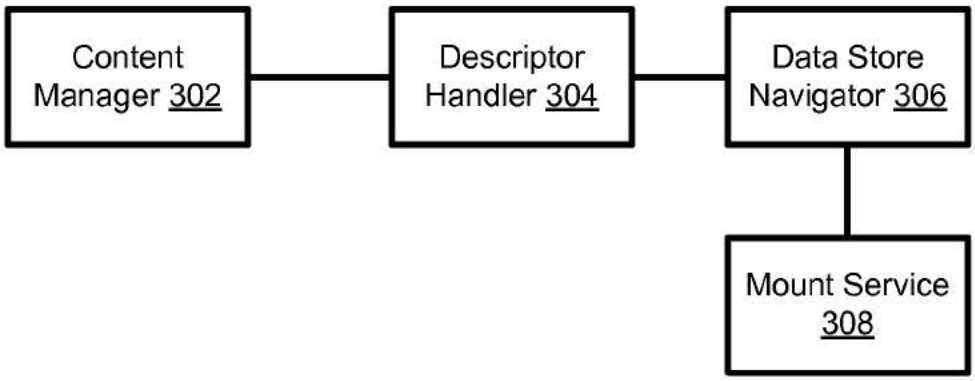

Content Manager 302
Descriptor Handler 304
Data Store Navigator 306
Mount Service 308

FIG. 3A

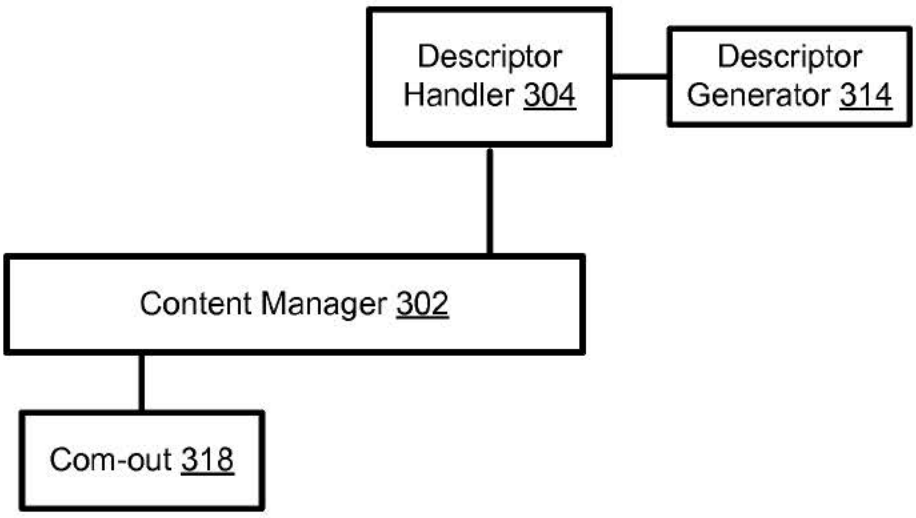

Descriptor Handler 304
Descriptor Generator 314
Content Manager 302
Com-out 318

FIG. 3B

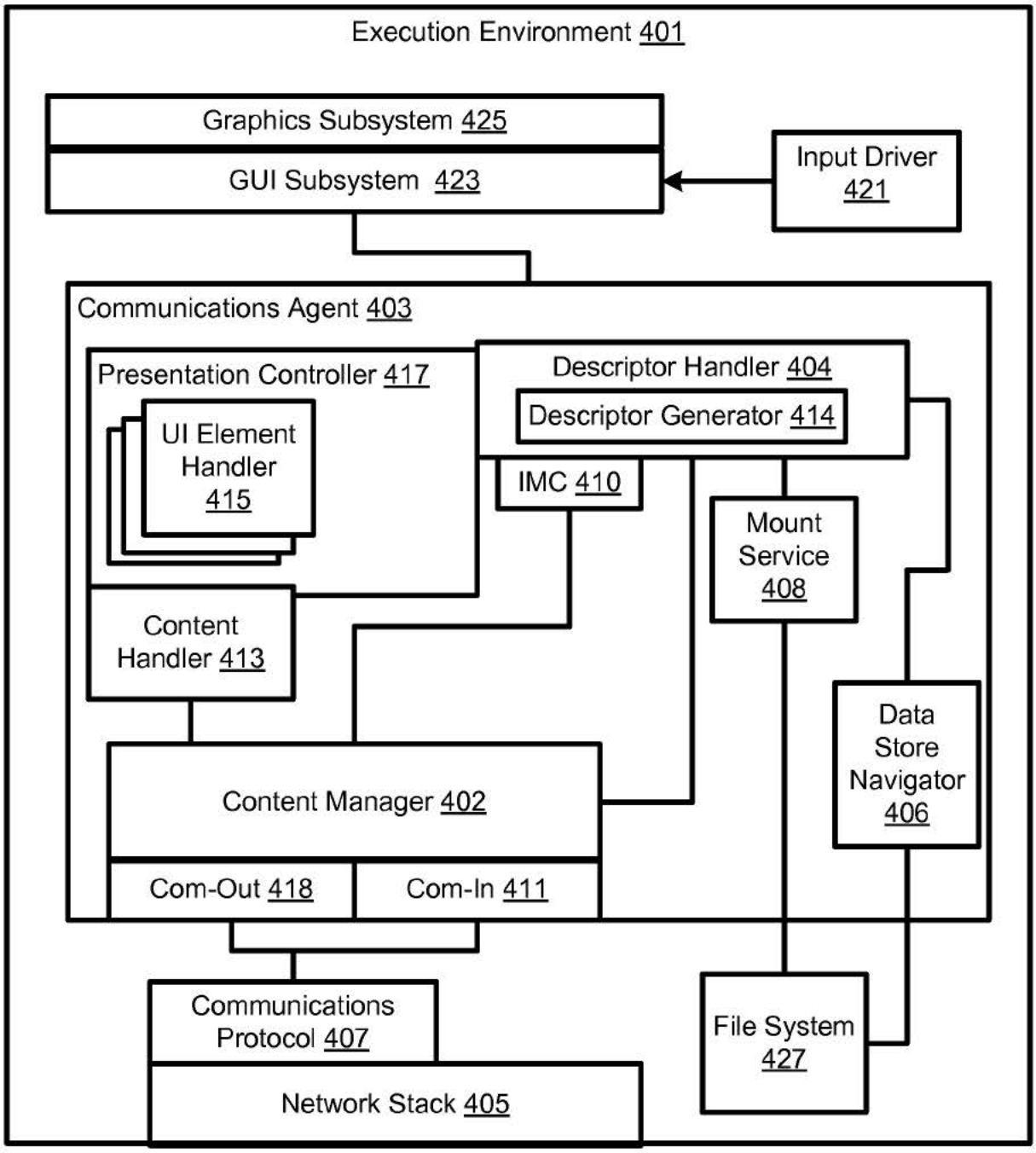

Execution Environment 401
Graphics Subsystem 425
GUI Subsystem 423
Input Driver 421
Communications Agent 403
Presentation Controller 417
UI Element Handler 415
Descriptor Handler 404
Descriptor Generator 414
IMC 410
Mount Service 408
Content Handler 413
Data Store Navigator 406
Content Manager 402
Com-Out 418
Com-In 411
Communications Protocol 407
Network Stack 405
File System 427

FIG. 4

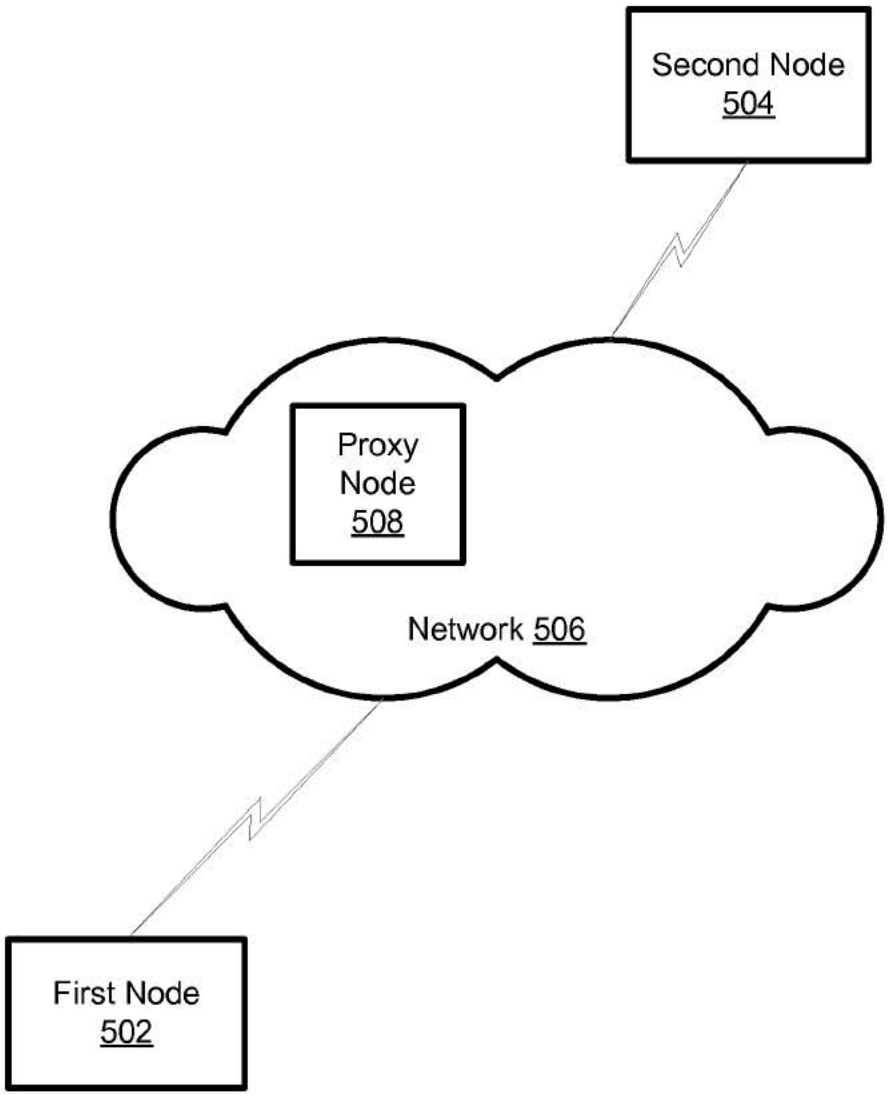


FIG. 5

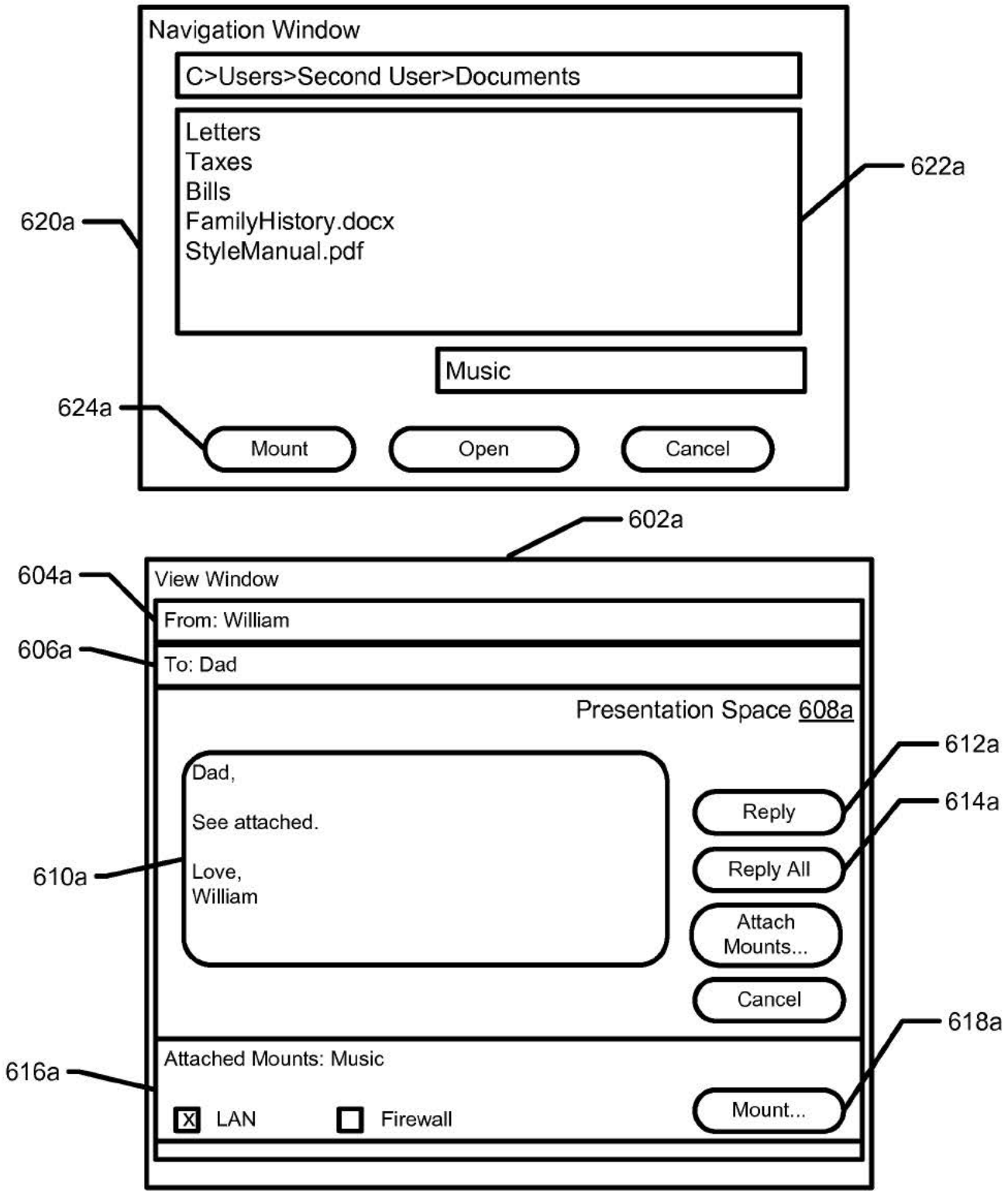

Navigation Window
C>Users>Second User>Documents
Letters
Taxes
Bills
FamilyHistory.docx
StyleManual.pdf
Music
Mount
Open
Cancel
620a
622a
624a
602a
604a
606a
View Window
From: William
To: Dad
Presentation Space 608a
Dad,
See attached.
Love,
William
Reply
Reply All
Attach Mounts...
Cancel
Attached Mounts: Music
LAN
Firewall
Mount...
610a
612a
614a
616a
618a

FIG. 6A

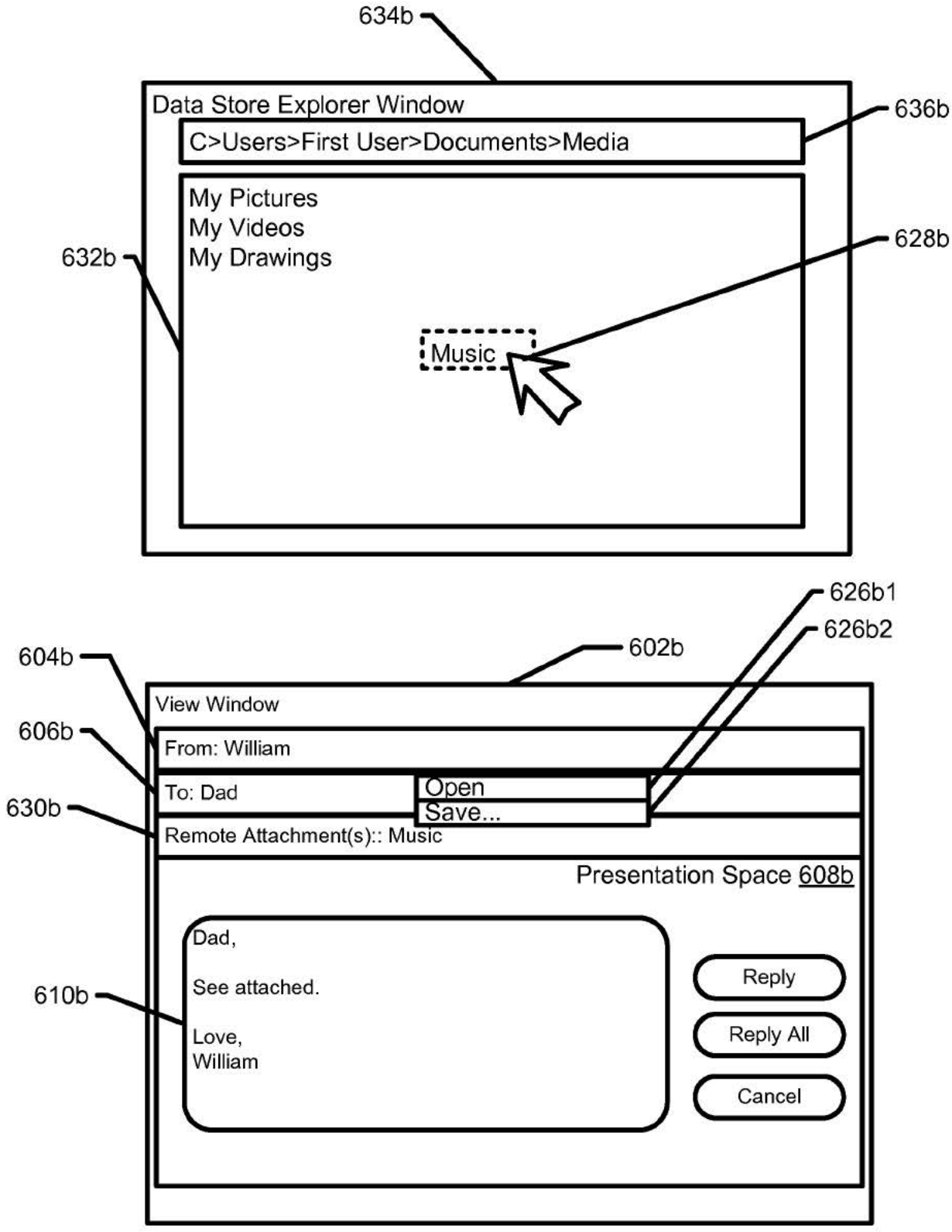

634b
Data Store Explorer Window
636b
C>Users>First User>Documents>Media
My Pictures
My Videos
My Drawings
632b
628b
Music
626b1
626b2
602b
604b
View Window
606b
From: William
To: Dad
Open
Save...
630b
Remote Attachment(s):: Music
Presentation Space 608b
Dad,
See attached.
Love,
William
610b
Reply
Reply All
Cancel

FIG. 6B

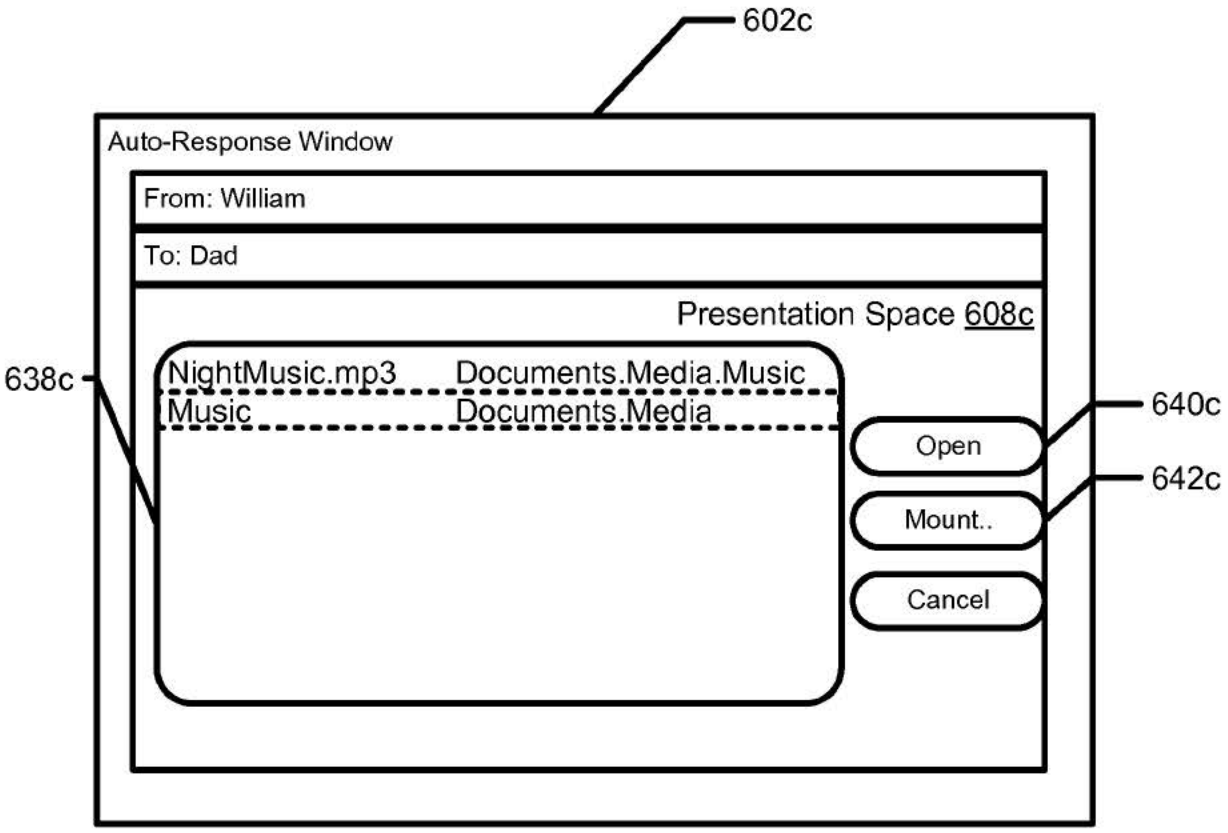

602c
Auto-Response Window
From: William
To: Dad
Presentation Space 608c
638c
NightMusic.mp3 Documents.Media.Music
Music Documents.Media
Open
Mount..
Cancel
640c
642c

FIG. 6C

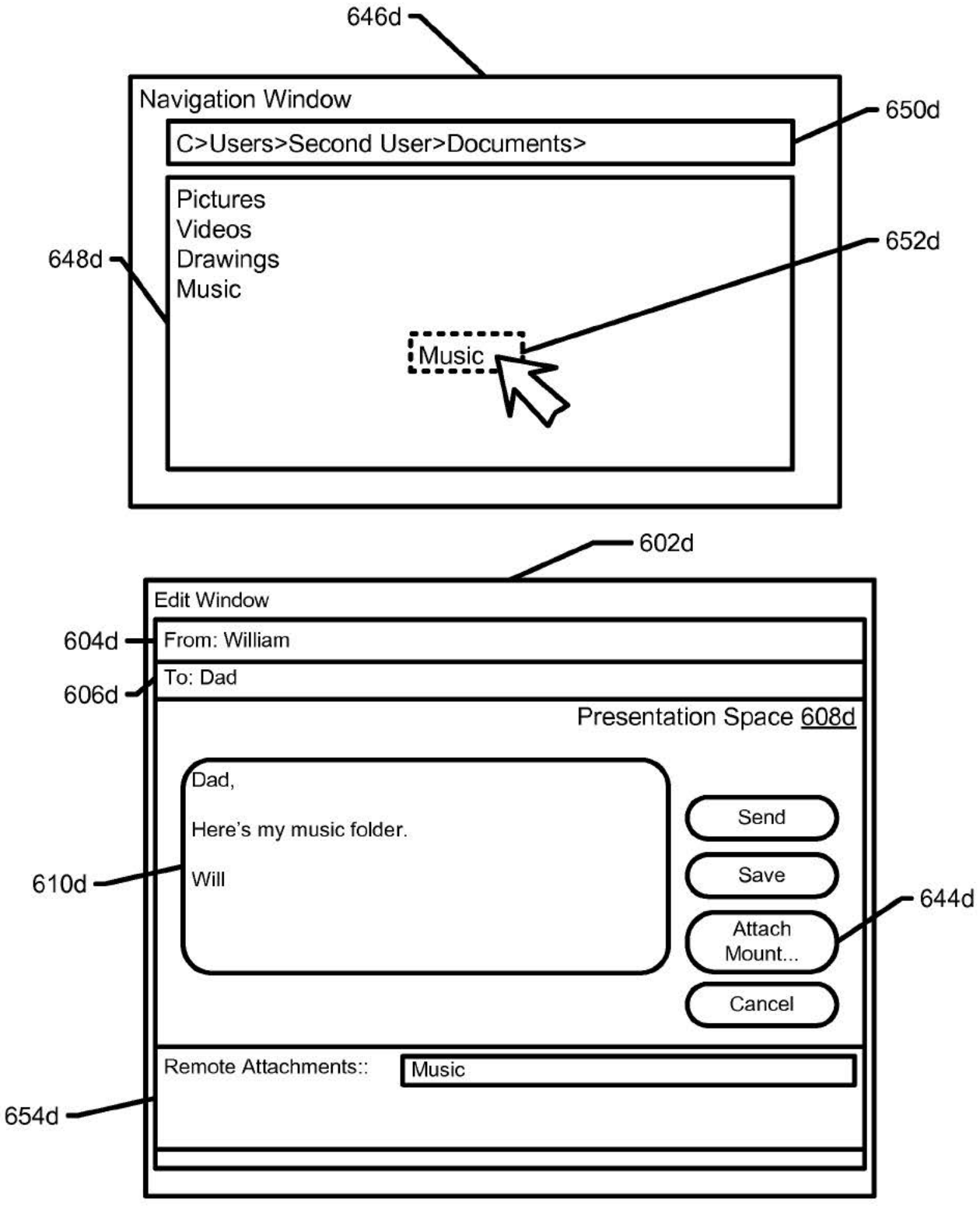

646d
Navigation Window
650d
C>Users>Second User>Documents>
Pictures
Videos
Drawings
Music
648d
652d
Music
602d
Edit Window
604d
From: William
606d
To: Dad
Presentation Space 608d
Dad,
Here's my music folder.
Will
610d
Send
Save
Attach Mount...
644d
Cancel
Remote Attachments::
Music
654d

FIG. 6D

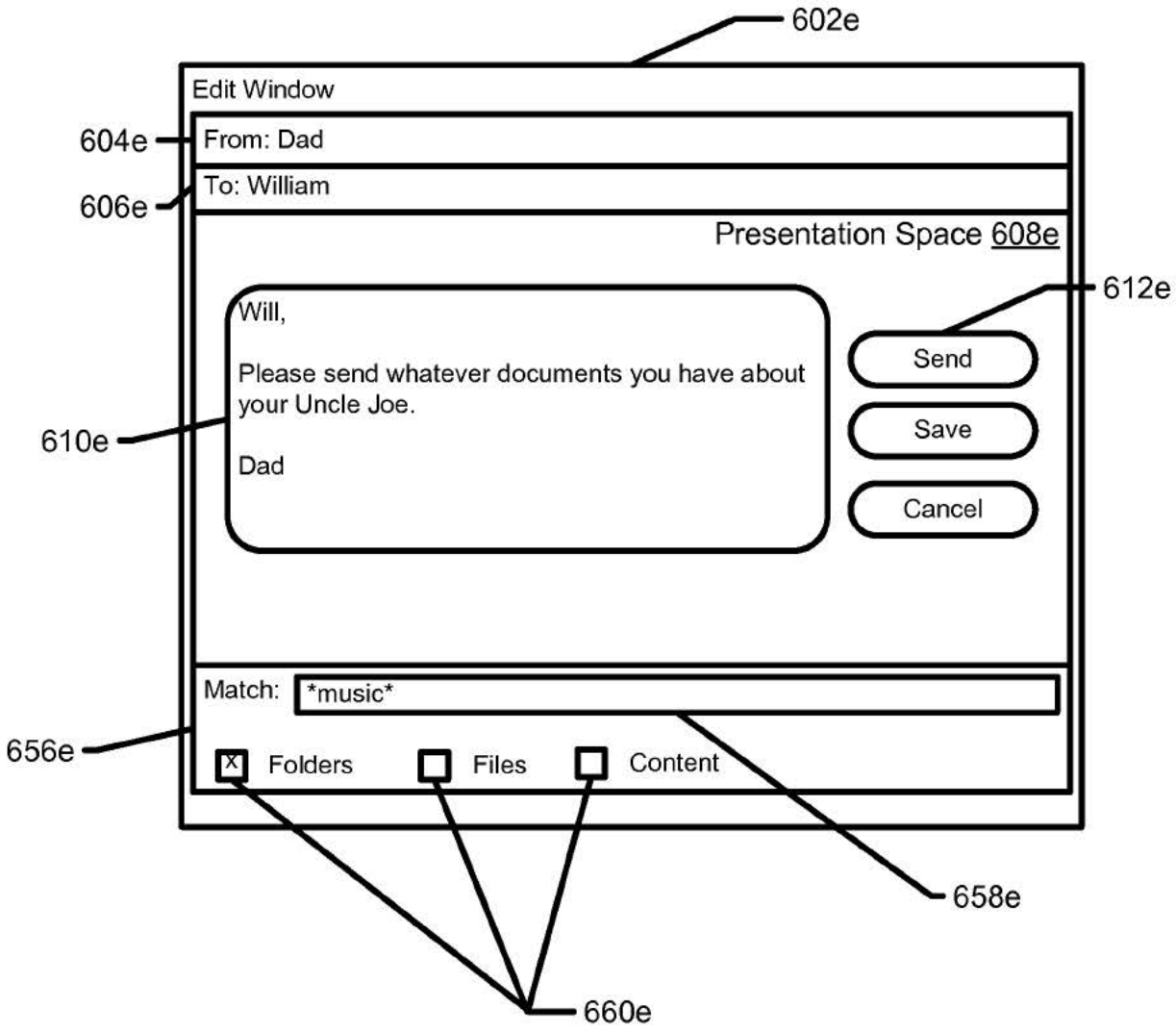

602e
Edit Window
604e
From: Dad
606e
To: William
Presentation Space 608e
612e
Will,
Please send whatever documents you have about your Uncle Joe.
610e
Dad
Send
Save
Cancel
Match:
music
656e
X
Folders
Files
Content
658e
660e

FIG. 6E

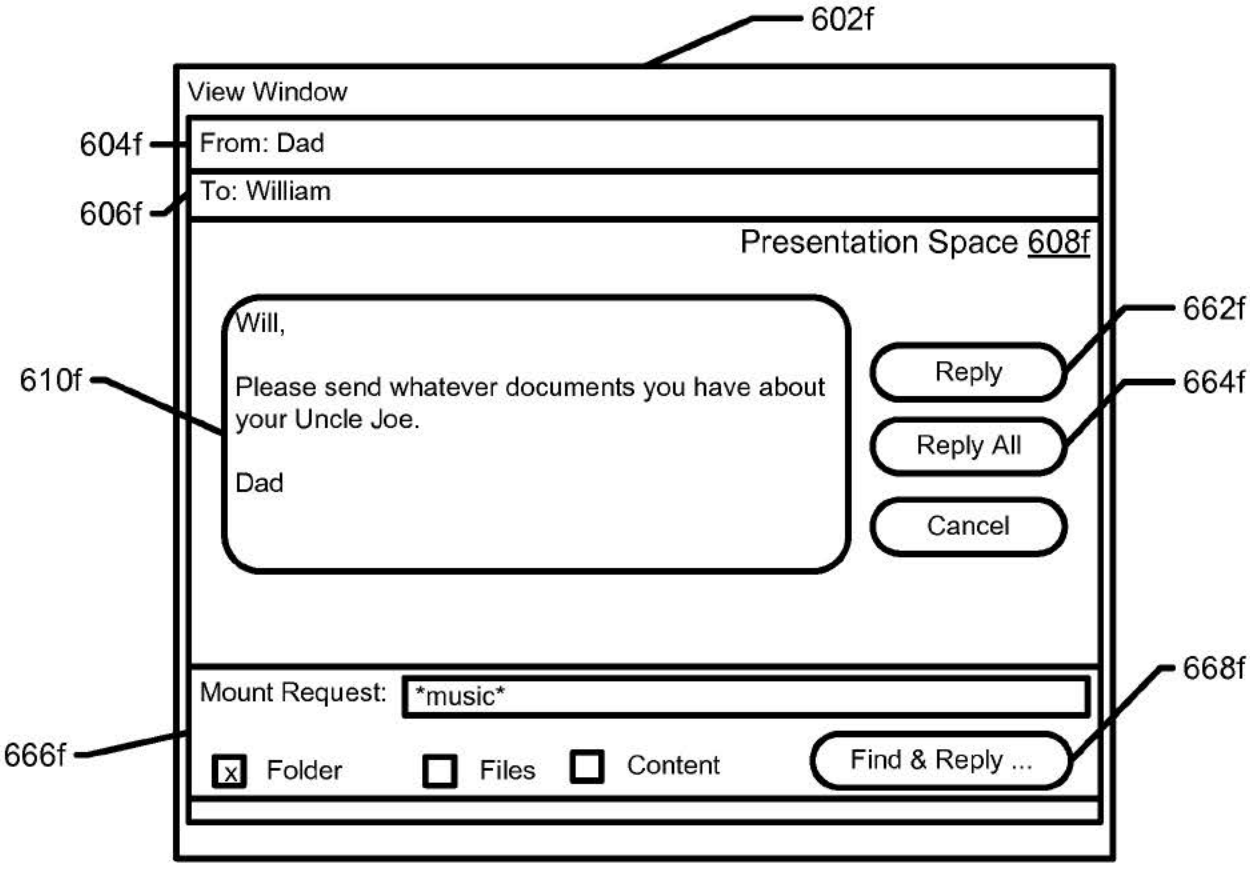

602f
View Window
604f
From: Dad
606f
To: William
Presentation Space 608f
610f
Will,
Please send whatever documents you have about your Uncle Joe.
Dad
662f
Reply
664f
Reply All
Cancel
668f
666f
Mount Request:
music
x
Folder
Files
Content
Find & Reply ...

FIG. 6F

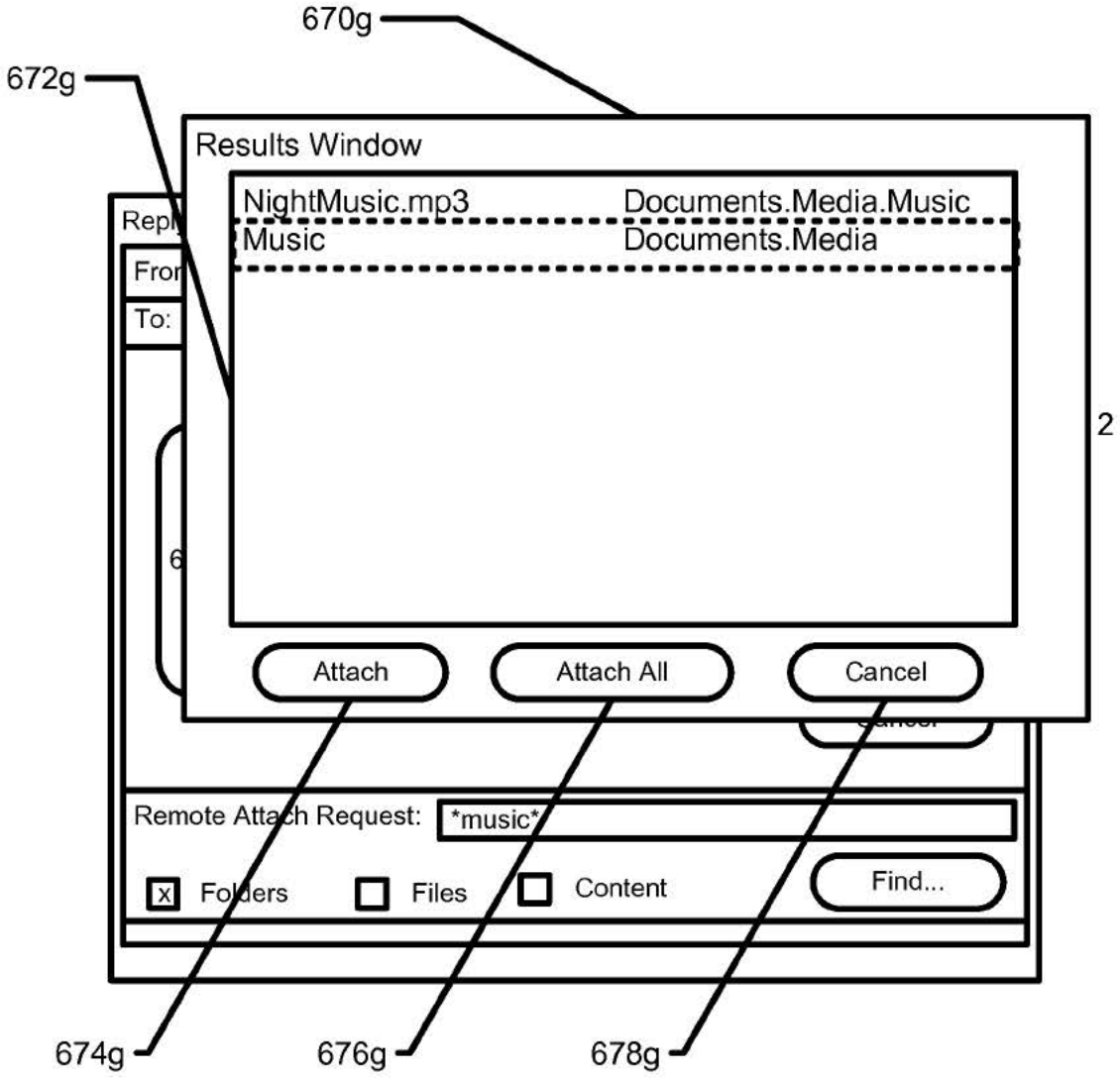

670g
672g
Results Window
NightMusic.mp3
Documents.Media.Music
Music
Documents.Media
Attach
Attach All
Cancel
Remote Attach Request:
music
Folders
Files
Content
Find...
674g
676g
678g

FIG.6G

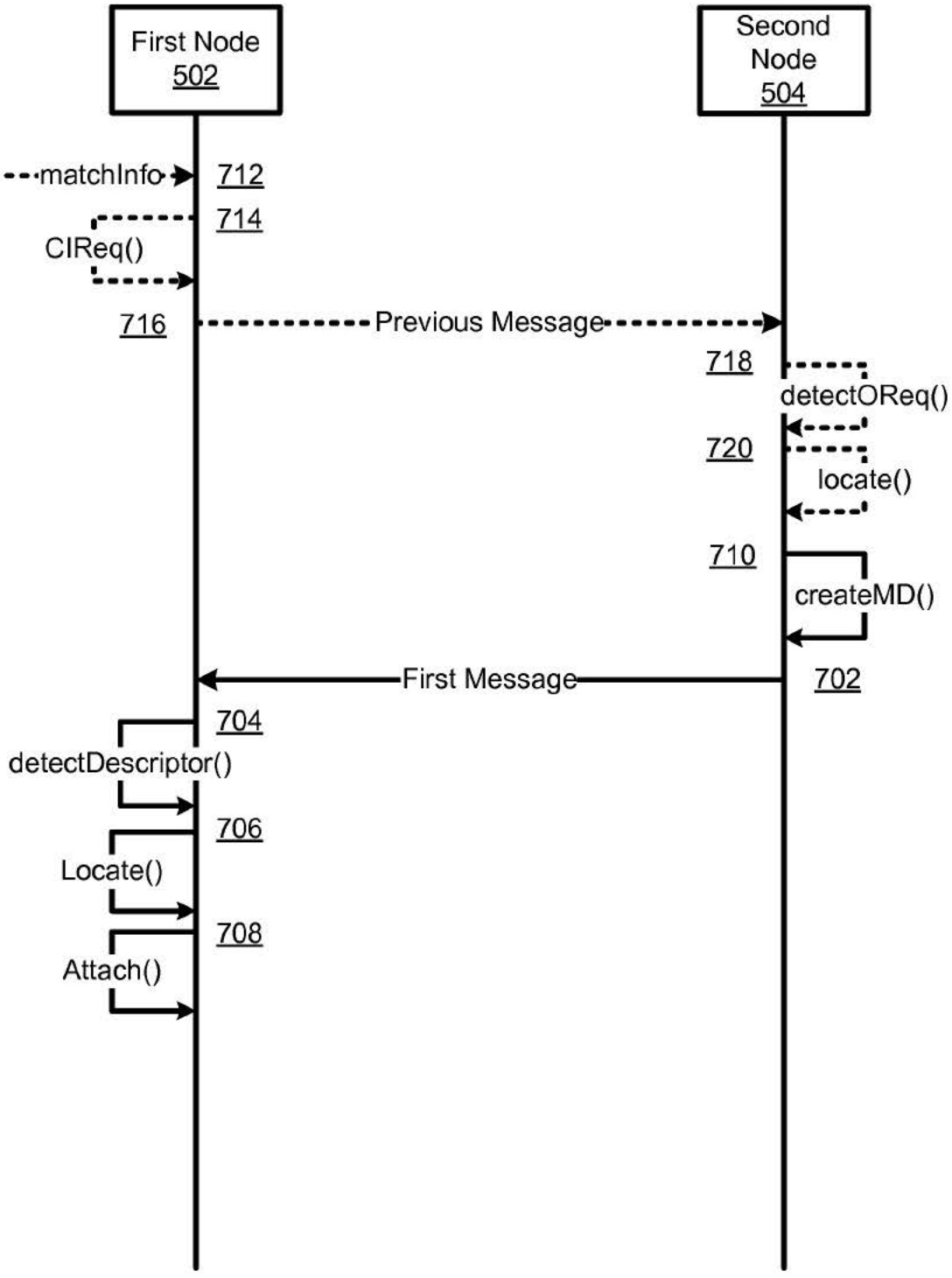

First Node
502
Second
Node
504
matchInfo
712
714
CIReq()
716
Previous Message
718
detectOReq()
720
locate()
710
createMD()
First Message
702
704
detectDescriptor()
706
Locate()
708
Attach()

FIG. 7

800a

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="--toOo----OoOt--------"

--toOo----OoOt---------
Content-Type: text/plain
Dad,

See attached.

Love,
William
--toOo----OoOt---------
Content-Type: application/mount-desscirptor

smb://SecondNode/Music
http://firstuser@secondnode.tld/users/seconduser/documents/media/music/
ftp://SecondNode/music/
--toOo----OoOt---------
```

802a 806a 808a 810a 804a 812a

Fig. 8A

800b

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="--OoOT----toOo--------"

--OoOT----toOo--------
Content-Type: text/plain
Hi, this is the first message
--OoOT----toOo--------
Content-Type: application/id-request
Accept:  image/*
<id-request>
  <and>
    <criterion attribute="query" type="keyword" match-expression="'*music*'" />
    <criterion attribute="scope" ressource-type="folder" />
  </and>
</remote-attach-requestt>
--OoOT----toOo--------
```

814b 812b 804b 806b 810a 808b 802b

FIG. 8B