

(19) World Intellectual Property
Organization
International Bureau



(43) International Publication Date
17 February 2005 (17.02.2005)

PCT

(10) International Publication Number
WO 2005/015432 A1

(51) International Patent Classification⁷: **G06F 17/30**,
17/60

(71) Applicant (for MG only): **IBM UNITED KINGDOM LIMITED** [GB/GB]; PO Box 41, North Harbour, Portsmouth, Hampshire PO6 3AU (GB).

(21) International Application Number:
PCT/EP2004/051698

(72) Inventors; and

(75) Inventors/Applicants (for US only): **BHOGAL, Kulvir, Singh** [US/US]; 137 Willow Ridge Road, Fort Worth, TX 76103-1054 (US). **ISHMAEL JR, Nizamudeen** [US/US]; 13331 Black Canyon Drive, Austin, TX 78729 (US). **KAMPER, Robert** [US/US]; 16835 Luckenwald Drive, Round Rock, TX 78681-5312 (US). **SAHASRABUDHE, Rohit** [US/US]; 3104 Lake Vista Drive, Louisville, KY 40241 (US). **SIDHU, Mandeep, Singh** [US/US]; 15216 Drusilla's Drive, Pflugerville, TX 78660 (US).

(22) International Filing Date: 3 August 2004 (03.08.2004)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
10/637,020 7 August 2003 (07.08.2003) US

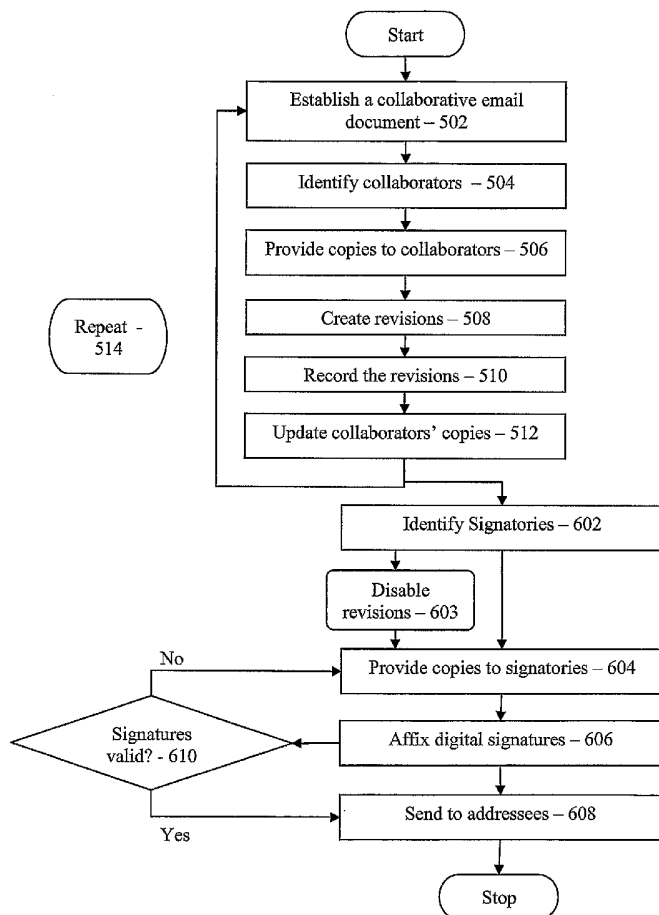
(74) Agent: **WILLIAMS, Julian**; IBM United Kingdom Limited, Intellectual Property Law, Hursley Park, Winchester, Hampshire SO21 2JN (GB).

(71) Applicant (for all designated States except US): **INTERNATIONAL BUSINESS MACHINES CORPORATION** [US/US]; New Orchard Road, Armonk, NY 10504 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM,

[Continued on next page]

(54) Title: COLLABORATIVE EMAIL



(57) Abstract: Methods, systems, and products are disclosed for writing collaborative email documents, including establishing a collaborative email document on an administrator's computer; identifying one or more collaborators who are authorized to view and edit the document; providing to the collaborators copies of the document for viewing and editing, wherein the collaborators' copies reside on collaborators' computers; creating revisions in at least one copy of the document; recording the revisions; and updating the copies of the document on collaborators' computers with the revisions. Embodiments typically include identifying editable portions of the email document, including specifying that only certain collaborators are authorized to view and edit one or more portions of the document. In many embodiments, revisions are streamed through a server so that there is no "master copy" of a collaborative document, on a server or elsewhere, against which revisions are recognized.



AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— with international search report

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM,

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Description

COLLABORATIVE EMAIL

Technical Field

- [001] The field of the invention is data processing, or, more specifically, methods, systems, and products for collaborative email.

Background Art

- [002] Systems for collaboration in developing email documents generally maintain a master copy of a document in a central location, record changes in the master copy, and update collaborators' copies by providing a new copy of the entire document. This uses a lot of bandwidth, particularly when there are many revisions over a period of time. There are version control systems, such as Unix's Source Code Control System or 'SCCS' and the open-source version control system known as the Concurrent Versions System or 'CVS.' Such systems are strongly oriented to version control for source code documents, however, and do not integrate very well with collaborative email, lacking, as they do, support for such collaborative features as automated updates to certain revision levels or authentication through valid digital signatures. For these reasons, there is an ongoing need for improvements in systems and methods for collaborative email.

Disclosure of Invention

- [003] In a first aspect, the present invention provides method for writing a collaborative email document, the method comprising: establishing a collaborative email document on an administrator's computer; identifying one or more collaborators who are authorized to view and edit the document; providing to the collaborators copies of the document for viewing and editing, wherein the collaborators' copies reside on collaborators' computers; creating revisions in at least one copy of the document; recording the revisions; and updating the copies of the document on collaborators' computers with the revisions.
- [004] Preferably, the collaborators include the administrator.
- [005] Preferably, establishing a collaborative email document on an administrator's computer comprises: identifying an administrator for the document; and retaining an administrative copy of the document on the administrator's computer.
- [006] The method preferably further comprises identifying editable portions of the email document, wherein identifying one or more collaborators further comprises specifying that only certain collaborators are authorized to view and edit one or more portions of the document.
- [007] The method preferably further comprises recording in the document authorization

for only certain collaborators to edit one or more portions of the document.

[008] The method preferably further comprises: identifying one or more signatories for the document; providing to each signatory a copy of the document for signing; affixing to the document a digital signature for each signatory; and sending the signed document from the administrator's computer to addressees.

[009] The method preferably further comprises sending the collaborative email document from the administrator's computer to addressees only when the document bears valid digital signatures from all signatories.

[010] Preferably, recording revisions comprises storing the revisions on a computer accessible to collaborators' computers.

[011] Preferably, recording revisions comprises storing the revisions with a version identification for each revision.

[012] Preferably, recording revisions comprises storing the revisions with a sequence identification for each revision.

[013] Preferably, updating the copies of the document with the revisions further comprises communicating the revisions asynchronously to the collaborators and to the administrator's computer.

[014] Preferably, updating the collaborators' copies of the document with the revisions further comprises communicating the revisions synchronously to one or more collaborators as the revisions are made.

[015] Preferably, updating the copies of the document with the revisions comprises: identifying collaborator computers that are available for synchronous communications of revisions; communicating revisions synchronously to collaborator computers that are available for synchronous communications; and updating current version identifications for the copies of the document on collaborator's computers that are available for synchronous communications.

[016] Preferably, identifying collaborator computers that are available for synchronous communications of revisions comprises determining whether collaborators copies are synchronized with the latest revisions.

[017] Preferably, updating the copies of the document with revisions comprises updating a copy of the document with all revisions later than a current version identifier for the copy.

[018] In a second aspect, the present invention provides a system for writing a collaborative email document, the system comprising: means for establishing a collaborative email document on an administrator's computer; means for identifying one or more collaborators who are authorized to view and edit the document; means for providing to the collaborators copies of the document for viewing and editing, wherein the collaborators' copies reside on collaborators' computers; means for creating

revisions in at least one copy of the document; means for recording the revisions; and means for updating the copies of the document on collaborators' computers with the revisions.

[019] Preferably, the collaborators include the administrator.

[020] Preferably, means for establishing a collaborative email document on an administrator's computer comprises: means for identifying an administrator for the document; and means for retaining an administrative copy of the document on the administrator's computer.

[021] The system preferably further comprises means for identifying editable portions of the email document, wherein means for identifying one or more collaborators further comprises means for specifying that only certain collaborators are authorized to view and edit one or more portions of the document.

[022] The system preferably further comprises means for recording in the document authorization for only certain collaborators to edit one or more portions of the document.

[023] The system preferably further comprises: means for identifying one or more signatories for the document; means for providing to each signatory a copy of the document for signing; means for affixing to the document a digital signature for each signatory; and means for sending the signed document from the administrator's computer to addressees.

[024] The system preferably further comprises means for sending the collaborative email document from the administrator's computer to addressees only when the document bears valid digital signatures from all signatories.

[025] Preferably, means for recording revisions comprises means for storing the revisions on a computer accessible to collaborators' computers.

[026] Preferably, means for recording revisions comprises means for storing the revisions with a version identification for each revision.

[027] Preferably, means for recording revisions comprises means for storing the revisions with a sequence identification for each revision.

[028] Preferably, means for updating the copies of the document with the revisions further comprises means for communicating the revisions asynchronously to the collaborators and to the administrator's computer.

[029] Preferably, means for updating the collaborators' copies of the document with the revisions further comprises means for communicating the revisions synchronously to one or more collaborators as the revisions are made.

[030] Preferably, means for updating the copies of the document with the revisions comprises: means for identifying collaborator computers that are available for synchronous communications of revisions; means for communicating revisions synchronously to collaborator computers that are available for synchronous commu-

nications; and means for updating current version identifications for the copies of the document on collaborator's computers that are available for synchronous communications.

[031] Preferably, means for identifying collaborator computers that are available for synchronous communications of revisions comprises means for determining whether collaborators copies are synchronized with the latest revisions.

[032] Preferably, means for updating the copies of the document with revisions comprises means for updating a copy of the document with all revisions later than a current version identifier for the copy.

[033] In a third aspect, the present invention comprises a computer program comprising computer program code to, when loaded into a computer system and executed thereon, cause said computer system to perform the steps of a method according to the first aspect.

[034] Preferred features of the computer program comprise code means to perform steps corresponding to the preferred features of the first aspect.

[035] The third aspect of the present invention may take the form of a computer program product for writing a collaborative email document, the computer program product comprising: a recording medium; means, recorded on the recording medium, for establishing a collaborative email document on an administrator's computer; means, recorded on the recording medium, for identifying one or more collaborators who are authorized to view and edit the document; means, recorded on the recording medium, for providing to the collaborators copies of the document for viewing and editing, wherein the collaborators' copies reside on collaborators' computers; means, recorded on the recording medium, for creating revisions in at least one copy of the document; means, recorded on the recording medium, for recording the revisions; and means, recorded on the recording medium, for updating the copies of the document on collaborators' computers with the revisions.

[036] Preferably, the collaborators include the administrator.

[037] Preferably, means for establishing a collaborative email document on an administrator's computer comprises: means, recorded on the recording medium, for identifying an administrator for the document; and means, recorded on the recording medium, for retaining an administrative copy of the document on the administrator's computer.

[038] The computer program product preferably further comprises means, recorded on the recording medium, for identifying editable portions of the email document, wherein means for identifying one or more collaborators further comprises means, recorded on the recording medium, for specifying that only certain collaborators are authorized to view and edit one or more portions of the document.

- [039] The computer program product preferably further comprises means, recorded on the recording medium, for recording in the document authorization for only certain collaborators to edit one or more portions of the document.
- [040] The computer program product preferably further comprises: means, recorded on the recording medium, for identifying one or more signatories for the document; means, recorded on the recording medium, for providing to each signatory a copy of the document for signing; means, recorded on the recording medium, for affixing to the document a digital signature for each signatory; and means, recorded on the recording medium, for sending the signed document from the administrator's computer to addressees.
- [041] The computer program product preferably further comprises means, recorded on the recording medium, for sending the collaborative email document from the administrator's computer to addressees only when the document bears valid digital signatures from all signatories.
- [042] Preferably, means for recording revisions comprises means, recorded on the recording medium, for storing the revisions on a computer accessible to collaborators' computers.
- [043] Preferably, means for recording revisions comprises means, recorded on the recording medium, for storing the revisions with a version identification for each revision.
- [044] Preferably, means for recording revisions comprises means, recorded on the recording medium, for storing the revisions with a sequence identification for each revision.
- [045] Preferably, means for updating the copies of the document with the revisions further comprises means, recorded on the recording medium, for communicating the revisions asynchronously to the collaborators and to the administrator's computer.
- [046] Preferably, means for updating the collaborators' copies of the document with the revisions further comprises means, recorded on the recording medium, for communicating the revisions synchronously to one or more collaborators as the revisions are made.
- [047] Preferably, means for updating the copies of the document with the revisions comprises: means, recorded on the recording medium, for identifying collaborator computers that are available for synchronous communications of revisions; means, recorded on the recording medium, for communicating revisions synchronously to collaborator computers that are available for synchronous communications; and means, recorded on the recording medium, for updating current version identifications for the copies of the document on collaborator's computers that are available for synchronous communications.

- [048] Preferably, means for identifying collaborator computers that are available for synchronous communications of revisions comprises means, recorded on the recording medium, for determining whether collaborators copies are synchronized with the latest revisions.
- [049] Preferably, means for updating the copies of the document with revisions comprises means, recorded on the recording medium, for updating a copy of the document with all revisions later than a current version identifier for the copy.
- [050] The present invention thus provides, in various aspects and preferred embodiments, methods, systems, and computer program products for writing a collaborative email document, including establishing a collaborative email document on an administrator's computer; identifying one or more collaborators who are authorized to view and edit the document, wherein the collaborators optionally include the administrator; providing to the collaborators copies of the document for viewing and editing, wherein the collaborators' copies reside on collaborators' computers; creating revisions in at least one copy of the document; recording the revisions; and updating the copies of the document on collaborators' computers with the revisions. In typical embodiments, establishing a collaborative email document on an administrator's computer includes identifying an administrator for the document and retaining an administrative copy of the document on the administrator's computer. In typical embodiments, updating copies of a document with revisions includes updating a copy of the document with one or more revisions later than a current version identifier for the copy.
- [051] Typical embodiments include identifying editable portions of the email document, wherein identifying one or more collaborators further comprises specifying that only certain collaborators are authorized to view and edit one or more portions of the document. Typical embodiments also include recording in the document authorization for only certain collaborators to edit one or more portions of the document.
- [052] Typical embodiments also include identifying one or more signatories for the document; providing to each signatory a copy of the document for signing; affixing to the document a digital signature for each signatory; and sending the signed document from the administrator's computer to addressees. Such embodiments typically also include sending the collaborative email document from the administrator's computer to addressees only when the document bears valid digital signatures from all signatories.
- [053] In typical embodiments, recording revisions comprises storing the revisions on a computer accessible to collaborators' computers. In such embodiments, recording revisions typically includes storing the revisions with a version identification for each revision. In such embodiments, recording revisions typically includes storing the revisions with a sequence identification for each revision.
- [054] In typical embodiments, updating copies of a collaborative email document with

revisions further comprises communicating the revisions asynchronously to the collaborators and to the administrator's computer. In many embodiments, updating the collaborators' copies of the document with the revisions further comprises communicating the revisions synchronously to one or more collaborators as the revisions are made. In such embodiments, updating the copies of the document with the revisions often includes identifying collaborator computers that are available for synchronous communications of revisions, communicating revisions synchronously to collaborator computers that are available for synchronous communications, and updating current version identifications for the copies of the document on collaborator's computers that are available for synchronous communications. In such embodiments, identifying collaborator computers that are available for synchronous communications of revisions typically includes determining whether collaborators copies are synchronized with the latest revisions.

Brief Description of the Drawings

- [055] Preferred embodiments of the present invention will now be described, by way of example only, with reference to the accompanying figures in which:
- [056] Figure 1 depicts an architecture for a data processing system in which various embodiments of the present invention may be implemented.
- [057] Figure 2 sets forth a block diagram of automated computing machinery useful in systems for collaborative email according to embodiments of the present invention.
- [058] Figure 3 sets forth a line drawing of a data entry screen on an email client improved according to embodiments of the present invention.
- [059] Figure 4 sets forth a database diagram of exemplary data structures useful in various embodiment of the present invention.
- [060] Figure 5 sets forth a flow chart depicting a method for writing a collaborative email document.
- [061] Figure 6 sets forth a flow chart illustrating an extension of the method of Figure 5.
- [062] Figure 7 sets forth a flow chart illustrating a method of synchronous communications of revisions.

Mode for the Invention

- [063] Introduction
- [064] The present invention is described to a large extent in this specification in terms of methods for collaborative email. Persons skilled in the art, however, will recognize that any computer system that includes suitable programming means for operating in accordance with the disclosed methods also falls well within the scope of the present invention.
- [065] Suitable programming means include any means for directing a computer system to

execute the steps of the method of the invention, including for example, systems comprised of processing units and arithmetic-logic circuits coupled to computer memory, which systems have the capability of storing in computer memory, which computer memory includes electronic circuits configured to store data and program instructions, programmed steps of the method of the invention for execution by a processing unit. The invention also may be embodied in a computer program product, such as a diskette or other recording medium, for use with any suitable data processing system.

[066] Embodiments of a computer program product may be implemented by use of any recording medium for machine-readable information, including magnetic media, optical media, or other suitable media. Persons skilled in the art will immediately recognize that any computer system having suitable programming means will be capable of executing the steps of the method of the invention as embodied in a program product. Persons skilled in the art will recognize immediately that, although most of the exemplary embodiments described in this specification are oriented to software installed and executing on computer hardware, nevertheless, alternative embodiments implemented as firmware or as hardware are well within the scope of the present invention.

[067] Definitions

[068] In this specification, the terms “field,” “data element,” and “attribute,” unless the context indicates otherwise, generally are used as synonyms, referring to individual elements of information, typically represented as digital data. Aggregates of data elements are referred to as “records” or “data structures.” Aggregates of records are referred to as “tables” or “files.” Aggregates of files or tables are referred to as “databases.” In the context of tables in databases, fields may be referred to as “columns,” and records may be referred to as “rows.” Complex data structures that include member methods, functions, or software routines as well as data elements are referred to as “classes.” Instances of classes are referred to as “objects” or “class objects.”

[069] “CGI” means “Common Gateway Interface,” a standard technology for data communications of resources between web servers and web clients. CGI provides a standard interface between servers and server-side ‘gateway’ programs that administer actual reads and writes of data to and from files systems and databases.

[070] “Client,” “client device,” “client machine,” or “client computer” means any computer or other automated computing machinery capable of administering collaborative email according to embodiments of the present invention. Examples include personal computers, PDAs, mobile telephones, laptop computers, handheld devices, and others as will occur to those of skill in the art. Clients include devices capable of

wireless as well as wireline communications.

[071] "HDML" stands for 'Handheld Device Markup Language,' a markup language used to format content for web-enabled mobile phones. HDML is proprietary to Openwave Systems, Inc., and can only be operated on phones that use Openwave browsers. Rather than WAP, HDML operates over Openwave's Handheld Device Transport Protocol ("HDTP").

[072] "HTML" stands for 'HyperText Markup Language,' a standard markup language for displaying web pages on browsers.

[073] "HTTP" stands for 'HyperText Transport Protocol,' a standard data communications protocol of the World Wide Web. HTTP is a hyperlinking protocol. In exemplary embodiments of the present invention, asynchronous communications of revisions are often implemented by use of hyperlinking protocols. Other examples of hyperlinking protocols include HDML and WAP.

[074] "POP" means Post Office Protocol, referring to the standard protocol for communicating email messages from email servers to email clients. "POP3" is a standard Post Office Protocol capable of communicating email messages among email servers and both to and from email clients, which means that POP3 is now useful as a single email protocol with no need for SMTP.

[075] "Server" in this specification refers to a computer or other automated computing machinery on a network that manages resources, including documents, and requests for access to such resources. A "web server," is a server that communicates with clients through data communications application programs, such as browsers or microbrowsers, by means of hyperlinking protocols such as HTTP, WAP, or HDTP, in order to manage and make available to networked computers documents, digital objects, and other resources. Servers generally include applications programs that accept data communications connections in order to service requests from clients by sending back responses. Any given computer or program therefore may be both a client and a server. The use of these terms in this disclosure refers primarily to the role being performed by the computer or program for a particular connection or coupling for data communications, rather than to the computer or program's capabilities in general.

[076] "SMTP" means Simple Mail Transfer Protocol, referring to the standard protocol for communicating email messages from email clients to email servers and from email servers to other email servers. It is typical in prior art that SMTP is used to communicate email messages from source email clients to mailbox locations, and POP is then used to communicate the email messages from mailboxes to destination email clients.

[077] "TCP/IP" refers to two layers of a standard OSI data communications protocol

stack. The network layer is implemented with the Internet Protocol, hence the initials 'IP.' And the transport layer is implemented with the Transport Control Protocol, referred to as 'TCP.' The two protocols are used together so frequently that they are often referred to as the TCP/IP suite, or, more simply, just 'TCP/IP.' TCP/IP is the standard data transport suite for the well-known world-wide network of computers called 'the Internet.'

- [078] "WAP" refers to the Wireless Application Protocol, a protocol for use with handheld wireless devices. Examples of wireless devices useful with WAP include mobile phones, pagers, two-way radios, hand-held computers, and PDAs. WAP supports many wireless networks, and WAP is supported by many operating systems. WAP supports HTML, XML, and particularly WML (the Wireless Markup Language), which is a language particularly designed for small screen and one-hand navigation without a keyboard or mouse. Operating systems specifically engineered for handheld devices include PalmOS, EPOC, Windows CE, FLEXOS, OS/9, and JavaOS. WAP devices that use displays and access the Internet run "microbrowsers." The microbrowsers use small file sizes that can accommodate the low memory constraints of handheld devices and the low-bandwidth constraints of wireless networks.

[079] Collaborative Email

- [080] Methods and systems according to the present invention generally implement collaborative email by identifying collaborators authorized to view and edit a collaborative email document as it is being written. Segments of the document may be identified that only particular collaborators are authorized to view and edit.
- [081] Revisions are typically streamed asynchronously (or synchronously) among collaborators' clients through one or more servers. Such methods and systems typically identify an administrator for the document on whose client is maintained an administrative copy of the document. The administrator may or may not be an authorized collaborator, but the administrator's copy of the document may be synchronized to a current version of the document, so that the administrator may provide administrative services regarding development of the document, administrative services such as monitoring document status, securing valid digital signatures from signatories, and finally transmitting the final version of the document to addressees.
- [082] In such systems, the addressees may or may not include the collaborators, and the signatories may or may not be collaborators. For example, a manager may assign the manager's secretary to create a collaborative document, type its initial contents, and act as administrator for development of the document. The manager may designate one or more executives as signatories, executives known to the manager as responsible for the subject area of the document. The manager may or may not be a signatory. The manager may designate collaborators who will receive copies of the document to view

and edit while the document is being written. In fact, it may be said that the collaborative edits or revisions to the document actually implement the writing of the document. The manager may or may not be listed as a collaborator.

[083] When the collaborators are finished editing the document, the administrator may email it to the signatories for digital signatures. After all required signatures are affixed to the document, the administrator may 'send' the document to its addressees. In email clients improved according to embodiments of the present invention, the email client's 'send' function typically is not enabled for sending to addressees until all required signatures are present. The send button on collaborators' clients typically is not enabled at all. The administrator's copy is updated with revisions just as are all the collaborators' copies. The only thing that distinguishes operation of the administrator's client from collaborators' email clients is that sending to addressees typically will eventually become enabled on the administrator's client (when the document is signed), but not so on collaborators' clients. Revisions typically are the only transmissions of data from collaborators' clients regarding collaborative email documents.

[084] Exemplary methods, system, and products for collaborative email are further explained with reference to the accompanying drawings, beginning with Figure 1. Figure 1 depicts an architecture for a data processing system in which various embodiments of the present invention may be implemented. The system of Figure 1 includes a number of computers coupled for data communications in networks. Each of the computers depicted may function as an email client or server according to embodiments of the present invention.

[085] The system of Figure 1 includes networks 102, 104. Networks in such systems may comprise LANs, WANs, intranets, internets, the Internet, webs, and the World Wide Web itself. Such networks comprise media that may be used to provide couplings for data communications between various devices and computers connected together within a distributed data processing system. Such networks may include permanent couplings, such as wire or fiber optic cables, or temporary couplings made through wireline telephone or wireless communications.

[086] In the example of Figure 1, server 128 and server 105 are connected to network 102. Storage unit 132 is connected to network 102 through server 105. Revisions records 134 identifying revisions to a collaborative email document according to embodiments of the present invention may be stored on storage units like storage unit 132. In addition, several exemplary client devices including a PDA 106, a workstation 108, and a mobile phone 110 are connected to network 102. Network-enabled mobile phone 110 connects to network 102 through wireless link 116, and PDA 106 connects to network 102 through wireless link 114. In the example of Figure 1, server 128

couples directly to client workstation 130 and network 104 (which may be a LAN), which incorporates wireless communication links supporting a wireless coupling to laptop computer 126 and wireline protocols supporting a wired coupling to client workstation 112. Each of these exemplary client devices may support email clients for collaborative email according to embodiments of the present invention.

[087] Client devices and servers in such distributed processing systems may be represented by a variety of computing devices, such as mainframes, personal computers, personal digital assistants, web-enabled mobile telephones, and so on. The particular servers and client devices illustrated in Figure 1 are for explanation, not for limitation. Data processing systems may include additional servers, clients, routers, other devices, and peer-to-peer architectures, not shown in Figure 1, as will occur to those of skill in the art. Networks in such distributed data processing systems may support many data communications protocols, TCP/IP, HTTP, WAP, HDTP, SMS, SMNT, POP, and others as will occur to those of skill in the art. Various embodiments of the present invention may be implemented on a variety of hardware platforms in addition to those illustrated in Figure 1. Figure 1 is presented as an example of a heterogeneous computing environment in which various embodiments of the present invention may be implemented, not as an architectural limitation of the present invention.

[088] Although exemplary embodiments in this disclosure are described generally in terms of client-server architectures, that is not a limitation of the present invention. On the contrary, various embodiments of the present invention may be implemented, for example, in peer-to-peer architectures and in other computer architectural arrangements as will occur to those of skill in the art.

[089] Figure 2 sets forth a block diagram of automated computing machinery comprising a computer 106, such as a client device, email client, or server, useful in systems for collaborative email according to embodiments of the present invention. The computer 106 of Figure 2 includes at least one computer processor 156 or 'CPU' as well as random access memory 168 ("RAM"). Stored in RAM 168 is an application program 152. Application programs useful in implementing collaborative email include Java servlets and CGI scripts running on servers and email clients on client machines. Also stored in RAM 168 is an operating system 154. Operating systems useful in computers according to embodiments of the present invention include Unix, Linux, Microsoft NTTM, and others as will occur to those of skill in the art.

[090] The computer 106 of Figure 2 includes computer memory 166 coupled through a system bus 160 to the processor 156 and to other components of the computer. Computer memory 166 may be implemented as a hard disk drive 170, optical disk drive 172, electrically erasable programmable read-only memory space (so-called

'EEPROM' or 'Flash' memory) 174, RAM drives (not shown), or as any other kind of computer memory as will occur to those of skill in the art.

[091] The example computer 106 of Figure 2 includes communications adapter 167 that implements connections for data communications 184 to other computers 182, servers or clients. Communications adapters implement the hardware level of data communications connections through which client computers and servers send data communications directly to one another and through networks. Examples of communications adapters include modems for wired dial-up connections, Ethernet (IEEE 802.3) adapters for wired LAN connections, and 802.11b adapters for wireless LAN connections.

[092] The example computer of Figure 2 includes one or more input/output interface adapters 178. Input/output interface adapters in computers implement user-oriented input/output through, for example, software drivers and computer hardware for controlling output to display devices 180 such as computer display screens, as well as user input from user input devices 181 such as keyboards and mice.

[093] Exemplary embodiments of the present invention are further explained with reference to Figure 3. Figure 3 sets forth a line drawing of a data entry screen on an email client improved according to embodiments of the present invention. The data entry screen of Figure 3 includes a title line 302 that displays the fact that the document under edit is a collaborative email document, the document identification ("465"), and the name of the email client ("Client Name"). In actual embodiments, the Client Name is the actual name of an email client such as Microsoft's OutlookTM or Qualcomm's EudoraTM.

[094] The data entry screen of Figure 3 includes a horizontal pull-down menu 304 containing the usual menu items such as 'File,' 'Edit,' 'View,' and so on. The entry screen of Figure 3 also includes a toolbar 306 that includes buttons titled 'Display Revisions,' 'Commit Revisions,' 'Version Back,' 'Version Forward,' 'Synch,' 'Sign,' and 'Send.' The email client of Figure 3 is programmed to display revisions, or not, in response to operation of the 'Display Revisions' button in the toolbar 306. In the example of Figure 3, the following text is displayed for collaborative email document 465:

[095] Hyperlinks are implemented in HTML documents by use of the
[096] hyperlink anchor element whose start and end tags are <a> and
[097] .

[098] In this example, 'Display Revisions' is set 'on,' and the email client therefore displays revisions representing deleting 'hyperlink' from row 2, column 1 and inserting 'anchor' at row 2, column 1. If 'Display Revisions' were set 'off,' the text under edit would be displayed as:

- [099] Hyperlinks are implemented in HTML documents by use of the
- [100] anchor element whose start and end tags are <a> and .
- [101] The email client of Figure 3 is programmed so that in response to operation of the 'Commit Revisions' button in the toolbar 306, the email client encodes a current set of revisions of the copy of the document under edit on the client and sends them to a server. The server may be a web server, for example, making the revisions available for asynchronous delivery to other clients, or the server may be an instant messaging server, delivering the revisions synchronously to available clients. The encoding may be implemented, for example, as follows, for the exemplary revision described just above:
- [102] d 2 1 hyperlink
- [103] i 2 1 anchor
- [104] where "d 2 1 hyperlink" encodes "delete the text 'hyperlink' beginning at row 2, column 1," and "i 2 1 anchor" encodes "insert the text 'anchor' beginning at row 2, column 1."
- [105] On a server, revisions so encoded may be stored in data structures such as those shown in Figure 4. Figure 4 shows a database diagram in which records in a document table 402 are related one-to-many to records in a collaborator copy table 410 which are in turn related one-to-many to records in a revision table 422. Each record in the document table 402 represents a collaborative email document having an administrative copy and one or more collaborators' copies. Each record in the document table includes a document identification field 404, an identification of the administrator for the document 406, and a status field 408 for the document. The status field may be used, for example, to note that the document has been created but not yet distributed to collaborators, is presently under edit by collaborators, or is disabled for editing while it is made available for signing by signatories.
- [106] Each record in the collaborator copy table 410 represents a copy of a collaborative email document provided to a collaborator for viewing and editing. Each record in the collaborator copy table 410 includes an identification 412 of the collaborator to whom a copy was provided. The collaborator identification 412 may be implemented as, for example, a collaborator's email address. Each collaborator copy record 410 also includes a document identification field 404 identifying the document a copy of which was provided to the collaborator.
- [107] Each collaborator copy record 410 in this example also includes an identification of the current version 414 of the document currently provided as a copy to the collaborator. Updating copies of collaborative email documents on collaborators' clients with revisions is typically carried out by updating a copy of the document with all revisions later than a current version identifier for the copy, and the repository of the

current version identifier for the copy is often a server-side data structure such as the one shown in Figure 4. Storing a copy's current version identification server-side generally means that an update request message from a client need specify only the collaborator identification 412 and the document identification 404. That is, the update request message need not necessarily include the current version number of the collaborative email on the client because the server tracks the version identification 414 for that collaborator's copy of that document and can infer it from the collaborator identification and the document identification. The server then is programmed to respond to such an update request message with a response message transmitting all the revisions later than the current revision number for the copy identified in the request message – and then updating the current version identification 414 to the current version number. This process is referred to generally in this specification as full synchronization, a process often supported in email clients improved according to embodiments of the present invention by a 'Synch' button, for example, such as the one shown in the exemplary toolbar 306 for the email client of Figure 3.

[108] An alternative request message requests less than a full synchronization, such as, for example, the revisions comprising just the next version, or the revisions comprising a previous version. That is, email clients according to embodiments of the present invention are typically programmed to operate in response to, for example, a 'Version Forward' button, such as the one shown in the example toolbar 306 on Figure 3, by transmitting to a server a request for the revisions comprising the next version. 'Next version' means the version just after the version currently identified in a server-side current version identification such as the one at reference 414 on Figure 4. The server then is programmed to respond to such an update request message with a response message transmitting only the revisions identified by the next version number for the copy identified in the request message – and then updating the current version identification 414 to the next version number. Once again, in this method, the email client does not need to advise the server of its current version; the server knows. The request message need only identify itself as a 'next version only please' message type and provide the collaborator identification 412 and the document identification 404 for the pertinent collaborative document.

[109] Similarly, email clients according to embodiments of the present invention are typically programmed to operate in response to a 'Version Back' button, such as the one shown in the example toolbar 306 on Figure 3, by transmitting to a server a request for the revisions comprising the previous version. 'Previous version' means the version just before the version currently identified in a server-side current version identification such as the one at reference 414 on Figure 4. The server then is programmed to respond to such an update request message with a response message transmitting

only the revisions identified by the previous version number for the copy identified in the request message – and then updating the current version identification 414 to the previous version number. Once again, in this method, the email client does not need to advise the server of its current version; the server knows. The request message need only identify itself as a ‘previous version only please’ message type and provide the collaborator identification 412 and the document identification 404 for the pertinent collaborative document.

[110] Each record in the revision table 422 represents a revision to a collaborative email document. In addition to the revision itself 420, which may be string-encoded as described above, each revision record 422 also includes a collaborator identification 412 of the collaborator who created a revision, a document identification 404 of the document for which the revision is intended, a revision identification 416, and a sequence identification 418 for ordering revisions within a version.

[111] As mentioned above, email clients according to embodiments of the present invention are typically programmed so that in response to operation of a ‘Commit Revisions’ button in a toolbar like the one shown at reference 306 on Figure 3, the email client encodes a current set of revisions of the copy of the document under edit on the client and sends them to a server. Such a set of revisions, upon arriving at a server, often then is given a revision identification code 416, which may be implemented, for example, as a sequential integer, 1, 2, 3, and so on. The version identification code (or ‘version number’) identifies a set of revisions from a particular collaborator, when used in correct order to update copies of the document, as implementing a particular version of the document.

[112] In typical embodiments, the email client may be programmed to provide a sequence number for each revision identifying the correct order in which the revisions are to be used to update copies of a collaborative email document. It is clear that the exemplary revisions mentioned above:

[113] d 2 1 hyperlink

[114] i 2 1 anchor

[115] could not be performed meaningfully in reverse order because after inserting ‘anchor’ at row 2, column 1, an email client could not then delete ‘hyperlink’ from the same location. This explains the usefulness of the sequence identification code shown at reference 418 on Figure 4.

[116] Methods of the invention are further explained with reference to Figure 5. Figure 5 sets forth a flow chart depicting a method for writing a collaborative email document, where the method includes establishing (502) a collaborative email document on an administrator’s client. The method of Figure 5 also includes identifying (504) one or more collaborators who are authorized to view and edit the document as it is being

written. The collaborators optionally include an administrator for the document.

[117] Establishing (502) a collaborative email document on an administrative client may be carried out by opening a new email document, identifying the document as a collaborative one, and identifying an administrator for the document. In typical embodiments, establishing a collaborative email document includes retaining an administrative copy of the document on an administrative client while the document is written by one or more collaborators.

[118] It is useful to understand that an administrative copy of a collaborative email document according to embodiments of the present invention is not a master document according to prior art. In fact, according to embodiments of the present invention, there is no 'master copy' of a collaborative document against which revisions are recognized. All copies, including an administrative copy, here have the same status and eligibility for viewing, editing, and updating. In fact, if an administrator is also an authorized collaborator, then all copies are identical, and the only thing that distinguishes the administrator's copy is that eventually its administrative client will acquire an enabled 'send' function, while none of the other clients ever will.

[119] In the present invention, there is no limitation regarding the initial contents of a collaborative email document. The user who creates a document, typically the user to be identified as the administrator for the document, may be an author, writing a first draft, so that the initial content is substantial. Alternatively, the administrator may be a clerk who creates on behalf of a supervisor a completely blank document to be written entirely by its collaborators.

[120] Identifying collaborators and administrators may be carried out by including identifiers for them in meta data in an email document implemented in HTML format, for example, as shown here:

[121] <HTML>
[122] <HEAD>
[123] <META name="Document Type"
[124] content="Collaborative Email">
[125] <META name="Revision Number" content="1">
[126] <META name="adminID"
[127] content="MaryJohnson@us.ibm.com">
[128] <META name="collaboratorID"
[129] content="MaryJohnson@us.ibm.com
[130] PeteJones@us.ibm.com JohnSmith@us.ibm.com">
[131] <META name="RevisionAuthority"
[132] content="MaryJohnson@us.ibm.com ALL
[133] PeteJones@us.ibm.com Segment1

[134] JohnSmith@us.ibm.com Segment2">

[135] </HEAD>

[136] <BODY>

[137] <SEGMENT1> </SEGMENT1>

[138] <SEGMENT2> </SEGMENT2>

[139] </BODY>

[140] </HTML>

[141] In this example, a metadata element named "Document Type" is used to identify an email document as a collaborative one. A metadata element named "Revision Number" is used to store the current revision identification of each copy of the document. The "Revision Number," although typically stored server-side also, may advantageously be recorded in the metadata for each copy of a collaborative document because each collaborator's copy and the administrative copy may embody different versions of the document at the same time.

[142] In the exemplary metadata, a metadata element named "adminID" records the identity of the administrator of the document as "MaryJohnson@us.imb.com." The administrator identification, although typically stored server-side also, may advantageously be recorded in the metadata for each copy of a collaborative document so that the email client where the copy is revised by a collaborator or an administrator can know whether that particular email client is the administrative client for that document.

[143] In this example, a metadata element named "collaboratorID" records the identities of the collaborators authorized to view and edit the document in the process of writing the document. The method of Figure 5 also includes providing (506) to the collaborators copies of the document for viewing and editing, and the collaborators' copies typically reside on collaborators' clients while the document is being written. In the metadata example, Mary Johnson, the administrator for the document, is also identified as a collaborator, which is optional. Providing copies to collaborators may be implemented by emailing copies from the administrative client to the collaborators. The 'From:' field (310 on Figure 3) in email clients according to embodiments of the present invention is often reserved for the identification of signatories, and communications of revisions among collaborators are carried out synchronously or asynchronously as described in this specification. It is possible, therefore, if an administrator is not listed as a collaborator, that none of the collaborators will ever know or care who the administrator is.

[144] The method of Figure 5 includes creating (508) revisions in at least one copy of the document. Revisions are created by recording changes in a document effected by the usual email client document editor and encoding them as described above: "d 5 12 hyperlink," "I 5 12 anchor," and so on, for example.

- [145] The method of Figure 5 includes recording (510) the revisions. Recording is often carried out by storing them temporarily on the email client until the collaborator making revisions arrives at a point where it is desirable to record them on the server as a version. In email clients according to embodiments of the present invention, a 'commit revisions' function is programmed to transmit to a server a set of revisions whose implementation represents a version. That is, in the method of Figure 5, recording revisions 510 may be implemented by storing the revisions on a server accessible to collaborators' clients. Many such email clients provide a toolbar button or other user interface control such as the 'Commit Revisions' button in the toolbar at reference 306 on figure 3 programmed to transmit to a server a set of revisions for a version. Such a transmission of revisions is not a normal email function of SMTP or POP, for example. Instead, according to embodiments of the present invention, such transmissions are effected by use of TCP/IP, HTTP, or some other data communications protocol as will occur to those of skill in the art.
- [146] In the method of Figure 5, recording revisions typically includes storing the revisions with a version identification for each revision. Storing revisions with versions identifications may be carried out by storing the version identifications in a server-side data structure such as that shown at reference 416 on Figure 4. Similarly, in the method of Figure 5, recording revisions often also includes storing the revisions with a sequence identification for each revision, so that the revisions can be sorted or indexed within a version. Storing revisions with sequence identifications may be carried out by storing the sequence identifications in a server-side data structure such as that shown at reference 418 on Figure 4. Persons of skill in the art will recognize that implementing a data structure like the revision table 422 of Figure 4 with a revision identification 416 and a sequence identification 418 creates a unique compound key into the revisions records comprising the revision identification and the sequence identification in combination. In such embodiments, the sequence identification may be a sequential integer, unique within a revision, non-unique across revisions, so that a unique key that includes the sequence identification also includes the revision number. Alternatively, the sequence identification may be a time stamp that identifies the date and time when each revision is made. Even a time stamp can be ambiguous across revisions, however, so that a unique key that includes a time stamp as sequence identification also advantageously includes the revision number.
- [147] The method of Figure 5 includes updating (512) the copies of the document on collaborators' clients with the revisions. Revisions created on a collaborator's client may be considered already incorporated into the document copy on that client, and the revision number on the client side may be updated accordingly, for example:
- [148] <META name="Revision Number" content="2">

- [149] Similarly, in response to the data communications transmission carrying the revisions of a version to a server, a server using server-side data structures like those of Figure 4 is programmed to update the pertinent collaborator copy record 410 with a new version number 414 for the collaborator copy where the new version was created. Such a server then creates a revision record 422 for each revision comprising the new version of the document.
- [150] Updating other collaborators' copies is then carried out by request/response communications between the collaborators' clients and the server. A collaborator client sends an HTTP request message for an update, either an update to the next version after the one presently installed on the client or a request for a full update all the way from the client's current version to the latest version available. A request for a full update all the way from the client's current version to the latest version available is called a synchronization request. The email client of Figure 3, for example, is programmed to transmit a request for a next version in response to operation of the 'Version Forward' button in its toolbar 306. The email client of Figure 3 is also programmed to transmit a request for a full update all the way from the client's current version to the latest version available in response to operation of the 'Synch' button in its toolbar 306.
- [151] In addition to requesting a next version and full synchronization, a client can also request a previous version. When a server receives such a request, the server provides in a response message the revisions defining a previous version, and the requesting email client implements those revisions in reverse order, thereby creating on that email client a previous version of the document. The email client of Figure 3, for example, is programmed to transmit a request for a previous version in response to operation of the 'Version Back' button in its toolbar 306.
- [152] Methods of collaborative email as illustrated in Figure 5 also often include identifying (516) editable portions of a collaborative email document. In such methods, identifying (504) one or more collaborators typically also includes specifying (518) that only certain collaborators are authorized to view and edit one or more portions of the document. Identifying (516) editable portions of a collaborative email document may be implemented by use of templates or by inserting markup identifying the editable portions. Moreover, authorization for only certain collaborators to edit one or more portions of the document advantageously may be recorded in the document itself – or in each copy of the document – so that such limitations on authorization are easily available to each collaborator's email client. In the following exemplary email document implemented in markup format, for example:

[153] <HTML>

[154] <HEAD>

[155] <META name="RevisionAuthority"
 [156] content="MaryJohnson@us.ibm.com ALL
 [157] PeteJones@us.ibm.com Segment1
 [158] JohnSmith@us.ibm.com Segment2">
 [159] </HEAD>
 [160] <BODY>
 [161] <SEGMENT1> </SEGMENT1>
 [162] <SEGMENT2> </SEGMENT2>
 [163] </BODY>
 [164] </HTML>,

[165] editable portions of a document are identified, in the document itself, by the markup elements <SEGMENT1> </SEGMENT1><SEGMENT2> </SEGMENT2>.

Specifying (518) that only certain collaborators are authorized to view and edit one or more portions of the document in this example is carried out by use of the metadata:

[166] <META name="RevisionAuthority"
 [167] content="MaryJohnson@us.ibm.com ALL
 [168] PeteJones@us.ibm.com Segment1
 [169] JohnSmith@us.ibm.com Segment2">,

[170] which specifies that Mary Johnson is authorized to view and edit the entire document, Pete Jones is authorized to view and edit Segment 1, and John Smith is authorized to view and edit Segment 2.

[171] Figure 6 sets forth a flow chart illustrating an extension of the method of Figure 5. That is, the method of Figure 6 includes establishing a collaborative email document 502, identifying collaborators 504, and so on, as in the method of Figure 5. In addition, however, the method of Figure 6 also includes identifying (602) one or more signatories for the document. Identifying the signatories may be implemented by programming an email client, such as the one shown in Figure 3, to accept as input through its 'From:' field 310 more than one entry of signatory identification, JaySchwarz@us.ibm.com, MikeWilliams@us.ibm.com, and so on, and treat each of them as a signatory. In such embodiments, there is no limitation regarding when the signatories are to be identified. That is, they may be identified when the document is first created by an administrator, or later, after all the collaborators are finished revising the document. In this sense, a collaborative email document according to the present invention is different from ordinary emails in that it may arrive in the in-boxes of collaborators with no entry in its 'From:' field, because, for example, no one has yet decided who must authorize the ultimate promulgation of the document to addressees. Consider an email announcement of a new corporate policy to be sent to hundreds of addressees. The email is created by a clerk in a corporate planning department and

revised by several collaborators. Then later, after the collaborators are satisfied with the contents of the document, it is decided whether the president of the company will sign it and whether one or more vice presidents will sign it.

[172] The method of Figure 6 includes providing (604) to each signatory a copy of the document for signing. To the extent that signatories are also collaborators, they may already have copy of the document, and they may affix their signatures as soon as revisions are concluded. Signatories who are not collaborators may be emailed a copy of the document, typically from the administrator, to obtain their signatures. In the email client of Figure 3, for example, the toolbar 306 includes a button labeled 'Sign,' where the email client is programmed to email the document to signatories. It is common in embodiments of this invention that the 'Sign' button, the ability to send a document to a signatory, is only enabled on the administrator's email client, so that there is administrative control over the process of affixing signatures to the document.

[173] The method of Figure 6 includes affixing (606) to the document a digital signature for each signatory. In this context, 'signature' generally refers to a digital signature. A digital signature is a data structure containing a hashed digest of the contents of a collaborative email document encrypted with a signatory's private key from a public/private key pair of an asymmetric cryptosystem. Email clients generally are equipped with the ability to affix digital signatures to email documents. In Microsoft OutlookTM, for example, a digital signature function is available, when an email document is open for editing, through the pull-down menus at View/Options/Security.

[174] Checking validity of a digital signature is accomplished by decrypting the hashed digest from the signature with a signatory's public key, hashing a new digest of the current contents of the document, and comparing the new digest with the decrypted digest from the signature. If the new digest and the decrypted digest from the signature are identical, the signature is considered valid. They will not be identical if revisions to the document are entered after the signature is affixed to the document.

[175] When the document is provided 604 to signatories for signing, therefore, in many embodiments of the present invention, the document advantageously is first disabled 603 for revision. A document may be disabled for revisions, for example, by encoding that fact in a data element dedicated to that purpose, such as, for example, the 'status' field 408 in the exemplary data structures of Figure 4. Such a status field may be given a value such as 'LOCKED,' for example, and a server supporting the revision table may then be programmed to exclude or reject revisions received from collaborators while the document is so locked.

[176] Some embodiments do not implement disabling of revisions during signing. In some embodiments that do implement disabling of revisions during signing, a signatory who is also a collaborator may go ahead and affix a signature that may be

rendered invalid by a later revision. In such embodiments, as shown in Figure 6, methods for collaborative email advantageously may include determining 610 whether the digital signatures in a document are valid, and, if they are not, providing 606 the document again to one or more signatories for re-signing. In this way, a method such as the one illustrated by Figure 6 may send a collaborative email document from an administrative client to addressees only when the document bears valid digital signatures from all signatories.

[177] The method of Figure 6 includes sending (608) the signed document from the administrative client to addressees. Unlike ordinary emails, the 'From:' field in a collaborative email arriving at addressees' in-boxes identifies the signatories of the document rather than its actual sender. The actual sender is typically an administrator who may or may not be known to the addressees. In this context, what the 'From:' field communicates is that the contents of the document are communicated to its addressees with the authority of the signatories.

[178] At this point in processing, the identities of the administrator and the collaborators may be of little concern to the addressees. The collaborators often in effect develop a document for mailing on behalf of signatories who have authority over the subject matter of the document. It is therefore the identity of the signatories rather than the identify of the developers of the document that will often be considered more pertinent to addressees. The collaborators, the administrator, and the signatories may or may not be among the addressees.

[179] Updating (512) copies of a collaborative email document with revisions is typically carried out by communicating the revisions asynchronously to the collaborators and to the administrative client. Asynchronous communication of revisions is useful because the collaborators' clients may not be on-line on the network to receive the revisions at the time when they are created. Asynchronous communication of revisions is carried out by use of server-side data structures such as those shown in Figure 4, for example. The email clients may be programmed to transmit the revisions in a standard message format such as HTTP (the HyperText Transport Protocol), WAP (the Wireless Access Protocol), HDTP (the Handheld Device Transport Protocol), or others as will occur to those of skill in the art. The server may support a Java Servlet or a CGI script, or other server-side functionality as will occur to those of skill in the art, for receiving the revisions and recording them in a database supporting data structures similar to the exemplary ones set forth in Figure 4. In this way, the revisions are stored on the server and remain available for download to collaborators' email clients when the clients go on-line on the network to retrieve the revisions for updating their copies of the documents.

[180] The exact method of updating a copy of a document on an email client may be

determined or selected by one or more setup parameters in the client. Updating a copy of a document by downloading revisions from a server may be carried out, for example, every time an email client is activated and at periodic intervals so long as the email client is operative. The periodic updates may be programmed to continue in background as long as the email client is running on a client machine even when the local copy of the document is not open for editing, so that when the copy opened for editing, it is likely to be fully synchronized with the latest revisions. Alternatively, a collaborator may wish to view the revisions since his or her last review of the document one at a time, stepping through them by use of, for example, a 'Version Forward' button like the one illustrated in the toolbar at reference 306 on the email client of Figure 3. In this case, the collaborator would turn off the email client's setup parameter for automatic background updates.

[181] In addition to asynchronous updates implemented through a server, updating (512) the collaborators' copies of a collaborative email document with revisions may be carried out by communicating the revisions synchronously to one or more collaborators as the revisions are made. Synchronous communications of revisions may be implemented by use of messages in an instant messaging protocol. Examples of instant messaging protocols useful with various embodiments of the present invention include the Instant Messaging and Presence Protocol ("IMPP") specified by the IMPP Working Group of the Internet Engineering Task Force and the Mitre Corporation's Simple Instant Messaging and Presence Service ("SIMP").

[182] Figure 7 sets forth a flow chart illustrating a method of synchronous communications of revisions. In the method of Figure 7, updating (512) copies of a collaborative email document with revisions includes identifying (702) collaborator clients that are available for synchronous communications of revisions. Instant messaging services generally define and support 'presence services' that provide indications whether an instant messaging client is on-line. Email clients capable of supporting synchronous communications of revisions according to embodiments of the present invention often are improved to support messaging and presence detection through one or more instant messaging protocols. The method of Figure 7 includes communicating (704) revisions synchronously to collaborator clients that are available for synchronous communications, typically carried out by transmitting the revisions in instant messages to all collaborators' email clients that are determined to be presently on-line and available for instant messaging.

[183] Updating copies of documents synchronously affects their version identifications. In the method of Figure 7, therefore, updating the collaborators' copies includes updating (706) current version identifications for the copies of the document on collaborator's clients that are available for synchronous communications. Updating

current version identifications may be implemented with HTTP messages, for example, from a client to a server that maintains server-side data structures like those shown in Figure 4. Such a server typically supports server-side functionality such as a Java servlet or a CGI script programmed to update version identifications in response to a message from a client bearing a collaborator identification 412 and a document identification 404.

- [184] By way of further explanation, it is noted that synchronous updates and asynchronous updates are advantageously employed together. When synchronous communications of revisions are employed for collaborators' copies that are not synchronized with the latest revisions, there is a risk that the asynchronous revisions will be applied to a copy out of proper sequence, thereby causing confusion in the copy of the document on that collaborator's client. Synchronous communications of revisions therefore are beneficially implemented for collaborators copies of a document after collaborators' copies are synchronized with the latest revisions from an asynchronous source of revisions.
- [185] In the method of Figure 7, for example, identifying (702) collaborator clients that are available for synchronous communications of revisions typically also includes determining whether collaborators copies are synchronized with the latest revisions. Determining whether collaborators copies are synchronized with the latest revisions may be accomplished by comparing a current version identification (414 on Figure 4) for each collaborator's copy of a document with the latest revision identification 416 in a revision table such as the one exemplified at reference 422 on Figure 4. To the extent that it is determined that the version of a collaborator's copy on an email client that is 'present,' available for synchronous communications of revisions, is not the current version of the document, that copy may be updated by applying all the revisions later than its current revision number – and then updating its current version identification 414 to the latest version number.
- [186] It will often be the case that some collaborators on a document engage in synchronous communications of revisions when other collaborators on the same document are unavailable for synchronous communications of revisions. Email clients according to embodiments of the present invention therefore often transmit all revisions for storage on a server for later asynchronous downloads by the other collaborators at their convenience. In this way, even a collaborator who begins a synchronous session with other collaborators and logs off in the middle of the session, can later obtain the intervening revisions by asynchronous download from the server. When such a collaborator logs off during synchronous communication of revisions, the email client advantageously records the current version identification for that collaborator's copy of the document in a data structure such as that shown at reference

414 in Figure 4, for example. When the collaborators log off at the end of synchronous communication of revisions, their email clients advantageously record the current version identification for each collaborator's copy of the document in a data structure such as the exemplary one shown at reference 414 in Figure 4.

[187] It will be clear to one of ordinary skill in the art from the foregoing description that many modifications and changes may be made in various embodiments of the present invention.

Claims

- [001] A method for writing a collaborative email document, the method comprising: establishing a collaborative email document on an administrator's computer; identifying one or more collaborators who are authorized to view and edit the document; providing to the collaborators copies of the document for viewing and editing, wherein the collaborators' copies reside on collaborators' computers; creating revisions in at least one copy of the document; recording the revisions; and updating the copies of the document on collaborators' computers with the revisions.
- [002] The method of claim 1 wherein establishing a collaborative email document on an administrator's computer comprises: identifying an administrator for the document; and retaining an administrative copy of the document on the administrator's computer.
- [003] The method of claim 1 or claim 2 further comprising recording in the document authorization for only certain collaborators to edit one or more portions of the document.
- [004] The method of any of claims 1 to 3 further comprising: identifying one or more signatories for the document; providing to each signatory a copy of the document for signing; affixing to the document a digital signature for each signatory; and sending the signed document from the administrator's computer to addressees.
- [005] The method of any of claims 1 to 4 further comprising sending the collaborative email document from the administrator's computer to addressees only when the document bears valid digital signatures from all signatories.
- [006] The method of any of claims 1 to 5 wherein recording revisions comprises one of: storing the revisions on a computer accessible to collaborators' computers; storing the revisions with a version identification for each revision; or storing the revisions with a sequence identification for each revision.
- [007] The method of any preceding claim wherein updating the copies of the document with the revisions comprises: identifying collaborator computers that are available for synchronous communications of revisions; communicating revisions synchronously to collaborator computers that are available for synchronous communications; and updating current version identifications for the copies of the document on collaborator's computers that are available for synchronous communications.
- [008] The method of any preceding claim wherein updating the copies of the document with revisions comprises updating a copy of the document with all revisions later than a current version identifier for the copy.

- [009] A system for writing a collaborative email document, the system comprising: means for establishing a collaborative email document on an administrator's computer; means for identifying one or more collaborators who are authorized to view and edit the document; means for providing to the collaborators copies of the document for viewing and editing, wherein the collaborators' copies reside on collaborators' computers; means for creating revisions in at least one copy of the document; means for recording the revisions; and means for updating the copies of the document on collaborators' computers with the revisions.
- [010] A computer program comprising computer program code to, when loaded into a computer system and executed thereon, cause said computer system to perform the steps of a method as claimed in any of claims 1 to 8.

1/7

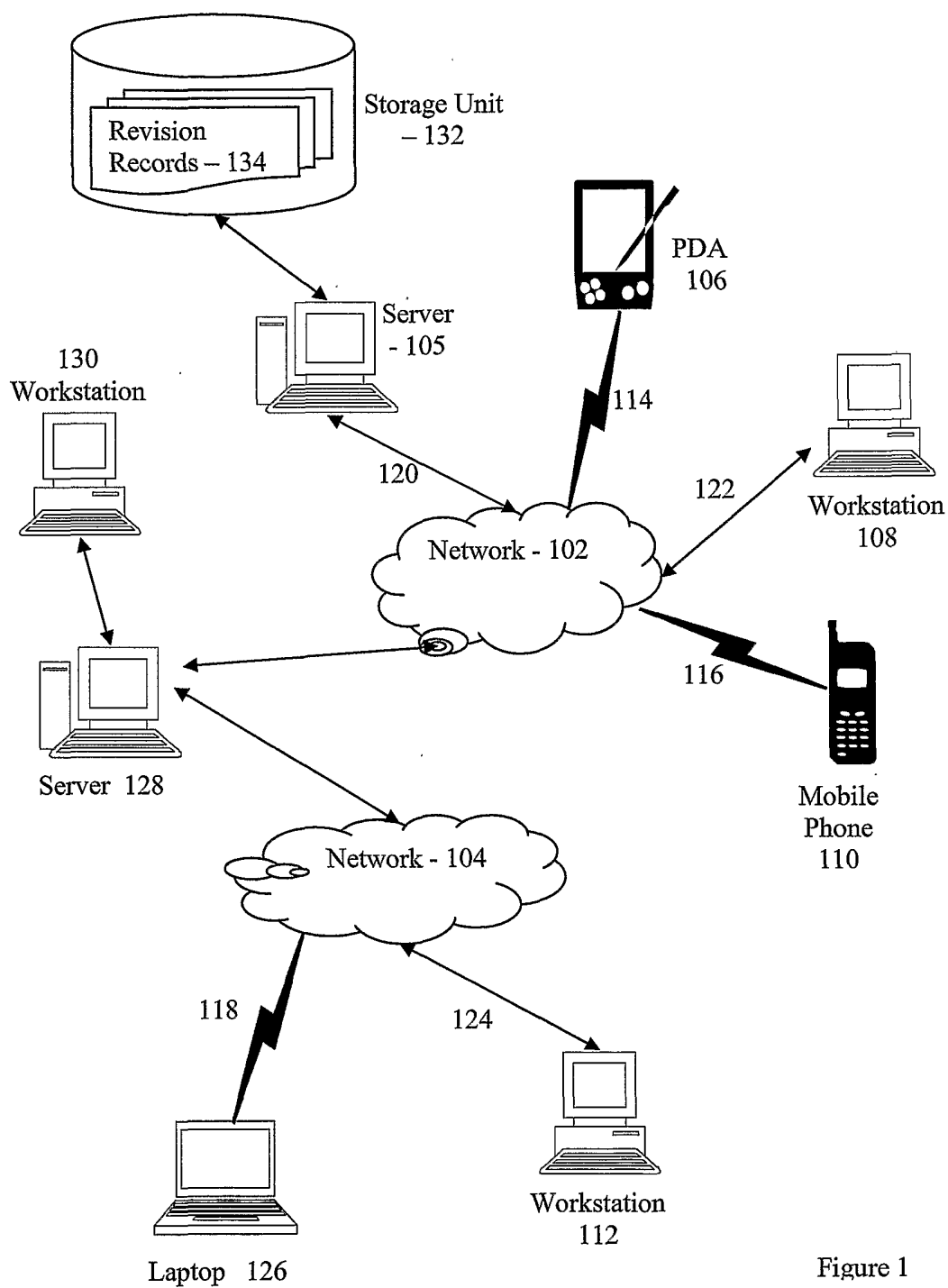


Figure 1

2/7

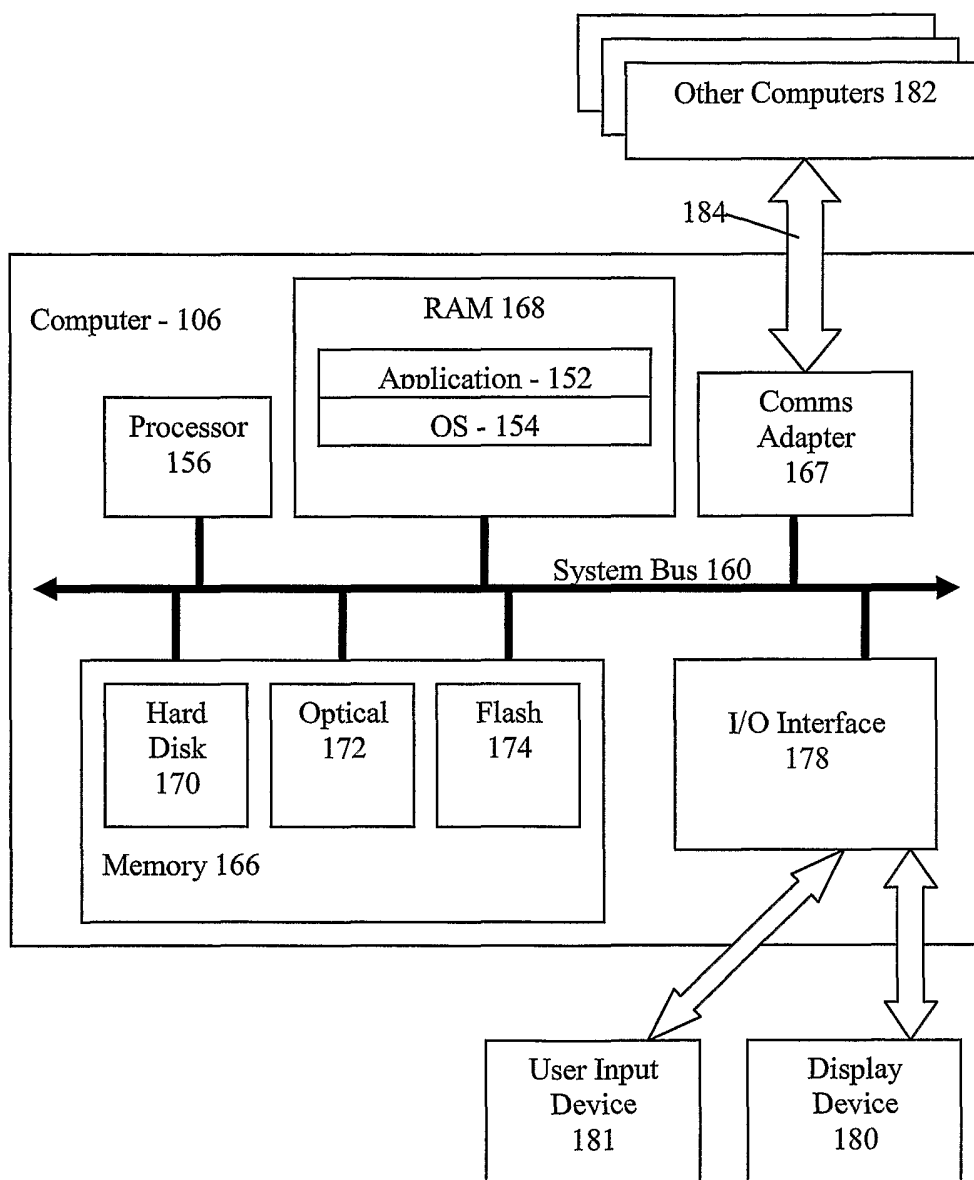


Figure 2

Collaborative Email – Document 465 – Client Name

File Edit View Tools Actions CollabOptions Help

Display Revisions

Commit Revisions

Version Back

Version Forward

Synch

Sign

Send

To:

From:

Collaborators:

Hyperlinks are implemented in HTML documents by use of the ~~hyperlink~~ anchor element whose start and end tags are <a> and .

← 302

← 304

← 306

← 308

← 310

← 312

Figure 3

4/7

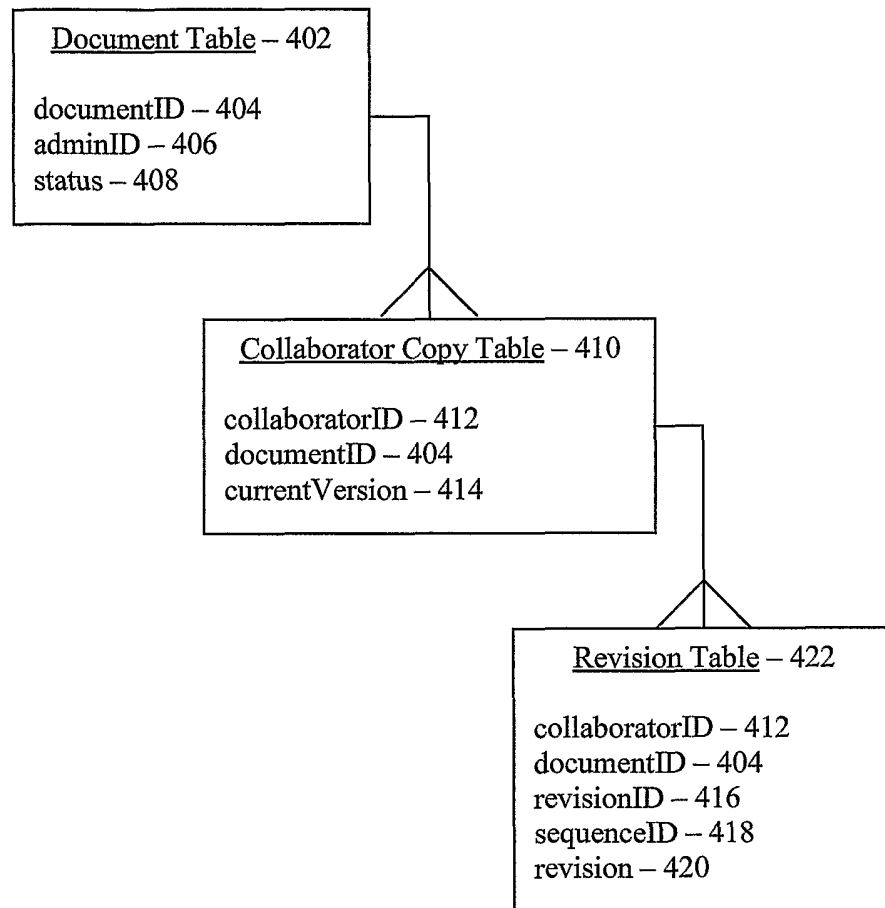


Figure 4

5/7

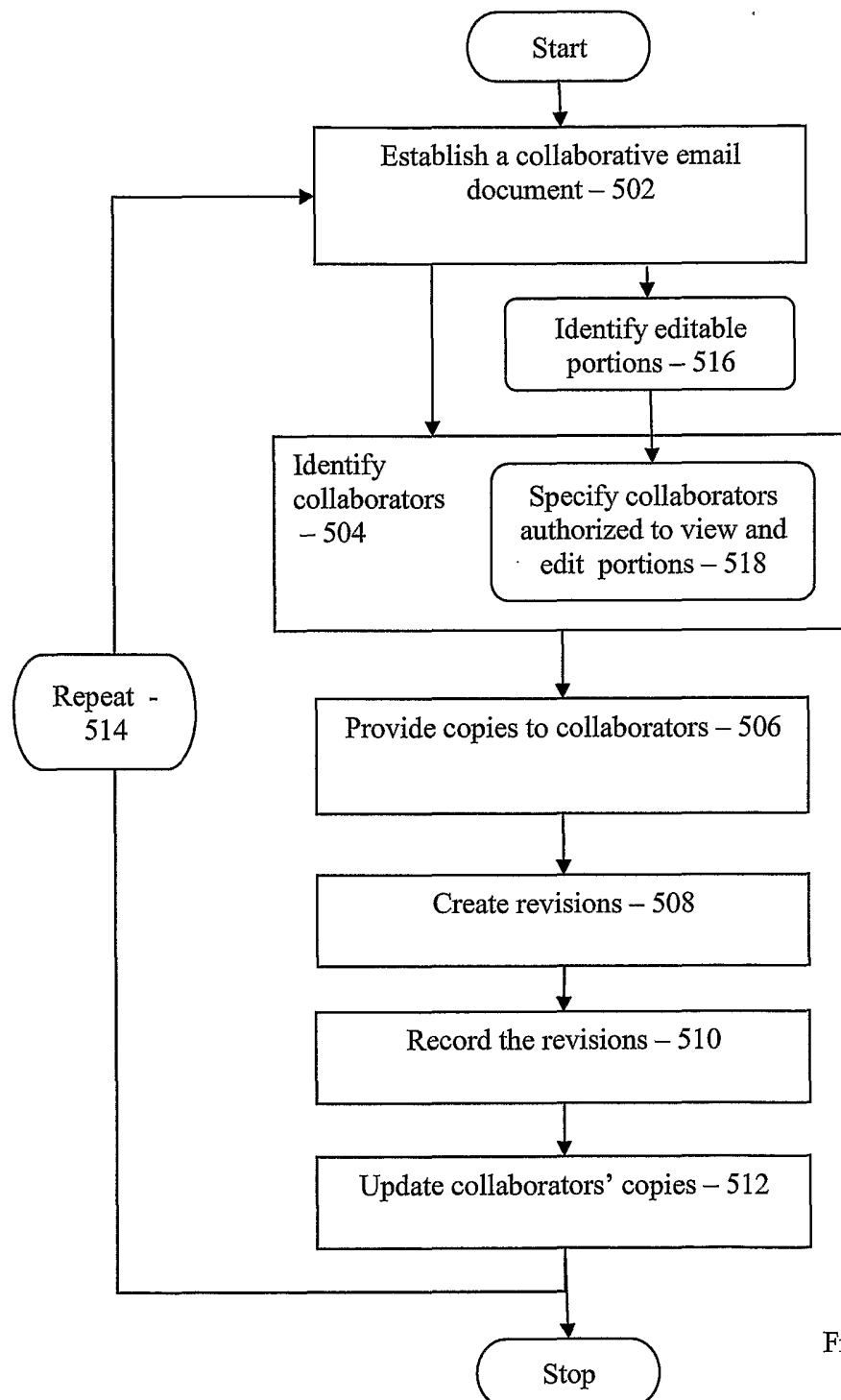


Figure 5

6/7

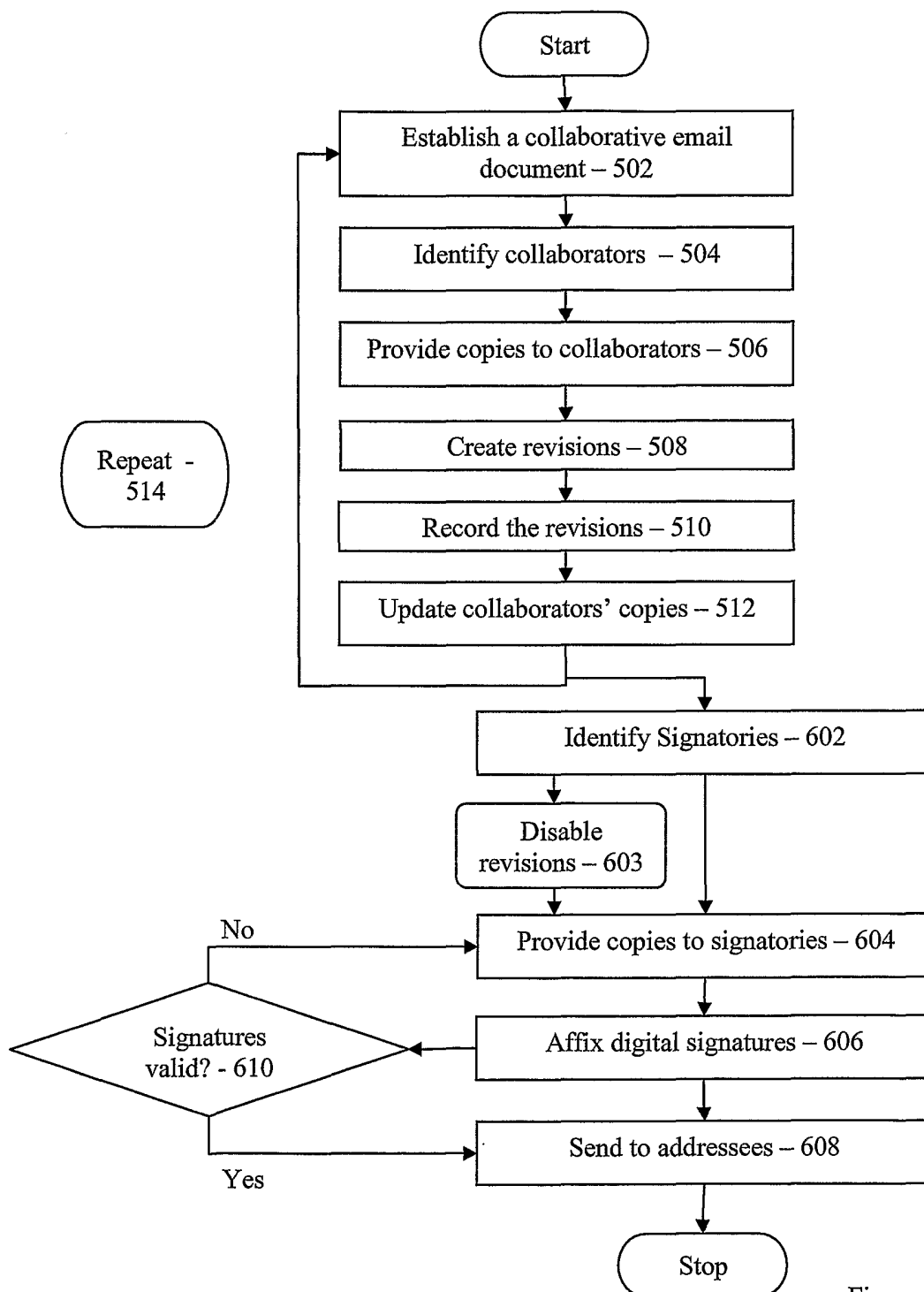


Figure 6

7/7

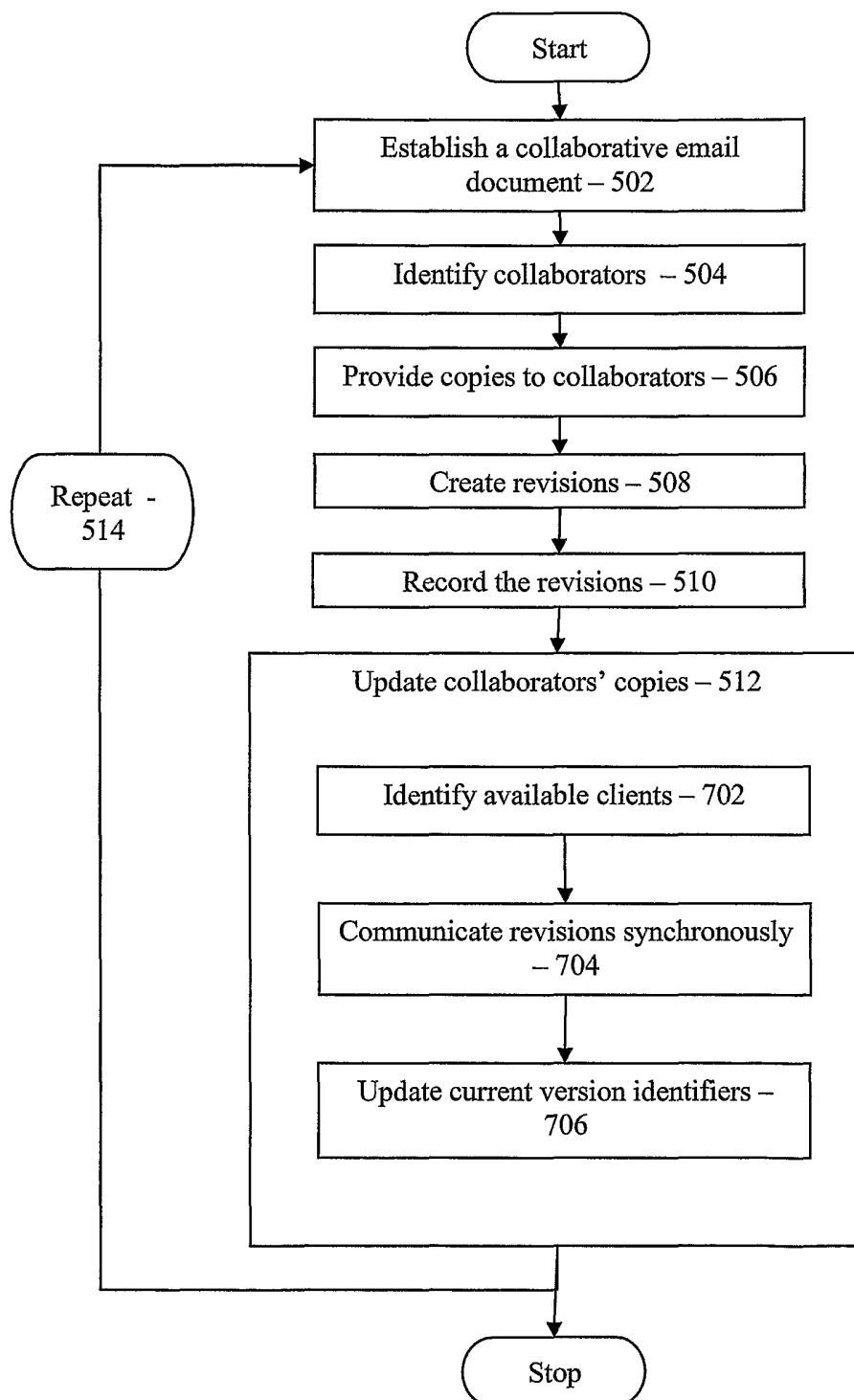


Figure 7

INTERNATIONAL SEARCH REPORT

International Application No
PCT/EP2004/051698

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F17/30 G06F17/60

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	PIER CEDERQVIST ET AL: "Version Management with CVS" 1993, SIGNUM SUPPORT AB, XP002299212 the whole document	1,2,6-10
Y	-----	3,4
Y	US 2003/050981 A1 (BANERJEE DWIP N ET AL) 13 March 2003 (2003-03-13) paragraphs '0030! - '0038!; compounds 1-24	3,4
A	----- US 2001/039553 A1 (LAMARCA ANTHONY G ET AL) 8 November 2001 (2001-11-08) paragraphs '0016!, '0031!, '0083!, '0106! -----	1-10

☐ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

° Special categories of cited documents:

- *A* document defining the general state of the art which is not considered to be of particular relevance
- *E* earlier document but published on or after the international filing date
- *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- *O* document referring to an oral disclosure, use, exhibition or other means
- *P* document published prior to the international filing date but later than the priority date claimed

- *T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- *X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- *Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
- * & * document member of the same patent family

Date of the actual completion of the international search

8 October 2004

Date of mailing of the international search report

03/11/2004

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Beker, H

INTERNATIONAL SEARCH REPORT

International Application No
PCT/EP2004/051698

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2003050981 A1	13-03-2003	NONE	
US 2001039553 A1	08-11-2001	US 6266682 B1 US 2003237051 A1	24-07-2001 25-12-2003