Title: METHOD, SYSTEM AND DEVICE FOR CONFLATING TIME-SENSITIVE MESSAGES

Abstract: Methods, devices and systems for conflating a time-sensitive message prior to transmission to a subscriber. A computer-implemented method to conflate a time-sensitive message prior to transmission of the time-sensitive message to a subscriber, the method comprising providing a queue comprising at least one time-sensitive message to be transmitted to a client; receiving a further time-sensitive message for transmission to the client; searching the queue for a time-sensitive message related to the further time-sensitive message; and if a related time-sensitive message is identified in the queue, generating a conflated message from the further time-sensitive message and the identified time-sensitive message.
METHOD, SYSTEM AND DEVICE FOR CONFLATING TIME-SENSITIVE MESSAGES

Background

Increasingly, the data that is transmitted over networks such as the Internet is of a time-sensitive nature. For example, the data may be continually changing or updating, as is the case with data relating to stock prices, currencies or the location of items in online games.

In a publish/subscribe system (known as a pub/sub system), a message broker manages the transmission of data. The message broker may be implemented on a server connected to the network. A subscriber indicates to the message broker which type or topic of data they are interested in. The publisher provides the data to the message broker, and the message broker provides the data to the interested subscribers. In some cases, the data received from the publisher may be changing so rapidly that once the message broker has received and processed the data it is already out of date. Accordingly, the message broker must have mechanisms in place to ensure that it is not providing out of date to interested subscribers.

The embodiments described below are not limited to implementations that solve any or all of the disadvantages of known publish/subscribe systems.

Summary

This Summary is provided to introduce a selection of concepts in a simplified form that are further described below in the Detailed Description. This Summary is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used as an aid in determining the scope of the claimed subject matter.

There is provided a computer-implemented method to conflate a time-sensitive message prior to transmission of the time-sensitive message to a subscriber, the method comprising providing a queue comprising at least one time-sensitive message to be transmitted to a client; receiving a further time-sensitive message for transmission to the client; searching the queue for a time-sensitive message related to the further time-sensitive message; and if a related time-sensitive message is identified in the queue, generating a conflated message from the further time-sensitive message and the identified time-sensitive message.

Searching the queue may comprise scanning each of the time-sensitive messages in the queue for time-sensitive messages related to the further time-sensitive message.

Searching the queue may comprise searching a lookup table for time-sensitive messages related to the further time-sensitive message, the lookup table comprising a list of topics associated with at least a portion of the time-sensitive messages in the queue.
The conflated message may comprise at least one of: the content of the time-sensitive message; the content of the further time-sensitive message; the content of both the time-sensitive message and the further time-sensitive message; and new content based on the content of the time-sensitive message and the further time-sensitive message.

The content of the conflated message may be based on an analysis of the content of the time-sensitive message and the further time-sensitive message.

The conflated message may be inserted in the queue at one of: the position of the time-sensitive message, and the end of the queue.

The conflated message may be inserted at the end of the queue, the method further comprises nullifying the time-sensitive message.

The conflated message may be inserted at the position of the time-sensitive message, the method further comprises nullifying the further time-sensitive message.

The conflated message may be inserted in a further queue or mailbox.

There is also provided a server to conflate a time-sensitive message prior to transmission of the time-sensitive message to a client, the server comprising a memory configured to store a queue comprising a plurality of time-sensitive messages to be sent to the subscriber; and a processor configured to: receive a further time-sensitive message; search the queue for a time-sensitive message related to the further time-sensitive message; if a time-sensitive message is located, generate a conflated message from the time-sensitive message and the further time-sensitive message.

Searching the queue may comprise scanning each of the time-sensitive messages in the queue for messages related to the further time-sensitive message.

Searching the queue may comprise searching a lookup table for time-sensitive messages related to the further time-sensitive message, the lookup table comprising a list of topics associated with at least a portion of the time-sensitive messages in the queue.

The conflated message may comprise at least one of: the content of the further time-sensitive message; the content of the time-sensitive message; the content of both the time-sensitive message and the further time-sensitive message; and new content based on the content of the time-sensitive message and the further time-sensitive message.

The content of the conflated message may be based on an analysis of the content of the time-sensitive message and the further time-sensitive message.
The conflated message may be inserted in the queue at one of: the position of the time-sensitive message; and the end of the queue.

The conflated message may be inserted at the end of the queue, the method further comprises nullifying the further time-sensitive message.

If the conflated message is inserted at the position of the time-sensitive message, the method may further comprise nullifying the further time-sensitive message.

The conflated message may be added to a further queue or mailbox.

The methods described herein may be performed by software in machine readable form on a tangible storage medium e.g. in the form of a computer program comprising computer program code means adapted to perform all the steps of any of the methods described herein when the program is run on a computer and where the computer program may be embodied on a computer readable medium. Examples of tangible (or non-transitory) storage media include disks, thumb drives, memory cards etc. and do not include propagated signals. The software can be suitable for execution on a parallel processor or a serial processor such that the method steps may be carried out in any suitable order, or simultaneously.

This acknowledges that firmware and software can be valuable, separately tradable commodities. It is intended to encompass software, which runs on or controls "dumb" or standard hardware, to carry out the desired functions. It is also intended to encompass software which "describes" or defines the configuration of hardware, such as HDL (hardware description language) software, as is used for designing silicon chips, or for configuring universal programmable chips, to carry out desired functions.

The preferred features may be combined as appropriate, as would be apparent to a skilled person, and may be combined with any of the aspects of the invention.

**Brief Description of the Drawings**

Embodiments of the invention will be described, by way of example, with reference to the following drawings, in which:

- Figure 1 is a schematic diagram of a publish/subscribe system for providing data to subscribers;
- Figure 2 is a schematic diagram of system for implementing conflation;
- Figure 3 shows a schematic diagram of a queue structure;
Figure 4 is a flowchart of a process for conflating a message; and

Figure 5 is a schematic diagram of a computing device.

Common reference numerals are used throughout the figures to indicate similar features.

Detailed Description

5 Embodiments of the present invention are described below by way of example only. These examples represent the best ways of putting the invention into practice that are currently known to the Applicant although they are not the only ways in which this could be achieved. The description sets forth the functions of the example and the sequence of steps for constructing and operating the example. However, the same or equivalent functions and sequences may be accomplished by different examples.

Reference is made to Figure 1, which illustrates a schematic diagram of a publish/subscribe system 100 for providing data to subscribers wherein data conflation is performed prior to providing the data to the subscribers. This prevents data from being sent to the subscribers that does not need to be sent. This may be because the data has been superseded and thus only the most recent data needs to be sent.

The publish/subscribe system 100 of Figure 1 comprises a first subscriber 102 operating a first client terminal 104, and a second subscriber 106 operating a second client terminal 108. It will be evident to a person of skill in the art that the principles described herein may also be applied to systems with more or fewer subscribers and/or client terminals. The client terminals 104 and 108 can be of any suitable form for receiving and presenting data to the subscribers. For example, these can be desktop, notebook or tablet computers, mobile telephones, or any other suitable electronic device able to receive transmitted data. In other examples, the client can also be in the form of a software application running on a suitable computing device.

25 The publish/subscribe system 100 also comprises a first publisher 110 and a second publisher 112. It will be evident to a person of skill in the art that the principles described herein may also be applied to systems with more or fewer publishers. The publishers 110 and 112 are publishing content of any type that is suitable for distribution over a publish/subscribe system. This includes, for example, stock prices, currency exchange prices, betting data or other financial information, news/sports items, multi-media data, and gaming data for online games. Some of the content provided by the publishers 110 and 112 is regularly or continuously updated.
The subscribers 102 and 106 can subscribe to one or more topics in the publish/subscribe system 100. This means that the subscribers 102 and 106 have requested to receive content of a certain type or relating to a certain subject. For example, the subscribers 102 and 106 can request to receive specific currency exchange prices or news items. In some examples, the subscribers 102 and 106 can subscribe to the same topic or different topics.

The client terminals 104 and 108 receive the subscribed content in the form of data messages sent over a communications network 114. In one example, the communications network 114 may be the Internet. However, in other examples, the communications network 114 can be a different network, such as a LAN, Wi-Fi access point, or private network, or comprise one or more additional networks, such as a mobile communication (cellular) network.

A server 116 is connected to the communication network 114 and configured to act as a message broker to manage and control the delivery of the publisher's content to the appropriate subscribers. More detail on the operation of the server 116 is outlined below with reference to Figures 2 to 5.

The server 116 comprises a storage device 118, which can be any device appropriate for storing data transiently or permanently. The content from the publishers is received in the form of data messages and the messages are stored at the storage device 118 in a storage device queue. As described in more detail below, the server 116 constructs a separate virtual queue 120 for each client terminal 104 and 108. The queues 120 are used to queue relevant data messages for transmission to the client terminals. In other words, if a data message is received from a publisher relating to a topic (e.g. a particular stock or sports league) to which one of the queues relates, then that data message is added to the virtual queue for that client terminal. However, the queues 120 are virtual in the sense that they do not contain the actual data messages themselves, but only a reference to where the data messages are stored on the storage device 118. In other words, the messages are not copied from the storage device to the queues, but only referenced from the storage device. This saves a significant amount of storage space, particularly when there are a large number of client terminals receiving data.

Messages in each queue are transmitted to the clients by a transmission system 122. In some examples, one transmission system 122 serves multiple virtual queues, although the number of queues served can be varied as appropriate, and similarly each queue can be served by more than one transmission system.

Messages in each virtual queue 120 may relate to the same topic and may relate to one another. For example, a first message may indicate a stock price has risen by 10 points, and a second message may indicate a stock price has fallen by 5 points. The cumulative effect of
these messages is that the stock price has risen by 5 points. Depending on the client’s preferences it may be possible to reduce the data transmitted to the client by sending a single message indicating the cumulative change of 5 points. This can be described as merging multiple delta or change values into one message. Alternatively data transmission may be made more efficient by using a single message to indicate the content of the two messages. For example a single message may indicate a rise and then a fall. Furthermore, there may be situations where a second message cancels out a first message and both messages can be deleted. Also, a second message may supersede a first message. For example, messages may contain absolute values with only the latest value being relevant. In this example it may be beneficial to only transmit the second message (either in the place of the first message or in the place of the second message).

The principles of reducing the amount of data transmitted to each client are implemented by the conflation system described hereinbelow.

Figure 2 shows a schematic diagram of a system for implementing conflation within the system of Figure 1. Messages are transferred from the storage device queue 118 to the client queues 120a, 120b as described previously. This transfer is performed by a multiplexer function 200a, 200b for each client queue 120a, 120b.

Multiplexers 200a, 200b transfer messages from the storage device queue 118 to client queues 201a, 201b for clients which have subscribed to the topic of each message.

During this transfer a conflation process is performed by comparing new messages to messages in the client queues 201a, 201b which have not yet been transmitted to the client.

The multiplexers 200a, 200b examine their respective queue for messages of the same topic as a new message being added to the queue. If a related message is found a conflation process is performed according to the rules activated for each topic. In various embodiments conflation may be switched on and off at different levels, for example at the topic or queue/client level. Furthermore, rules may be applied at different levels, for example at the topic or queue/client level. Each topic may have different rules specifying how the conflation operates. The rules may be defined by the system, or provision may be provided to enable user-definition of the rules. For example:

- A new message and the existing message may be combined into a single message in the queue. The data of each message may be included in the message (for example an indication that a price is risen and then falling), or the data may be merged to be an indication of the cumulative effect of the two message (for example if a first message indicates a rise of 10, and a second message indicates a fall of 5, the
conflated message may indicate a rise of 5). The conflated message may replace the existing message in the queue which maintains time and position in the queue, or appended to the end of the queue. Where messages are modified by the conflation process, the result may be cached such that the result can be re-used without requiring a full recalculation.

- The new message may replace the old message in the queue - for example if messages contain absolute values (rather than changes) it may be more appropriate to only transmit the latest value.

- The existing message may be deleted if it is superseded by the new message. A null value may be left in place of the old message, or it may be deleted completely.

- Both messages may be transmitted without editing.

Messages in the queue are transmitted to the client by a transmission system in a conventional way. This transmission may be managed and/or performed by the multiplexer so that it can be performed in coordination with the conflation process such that maximum benefit is derived from the conflation. Furthermore, the transfer of messages from the storage device queue to the client queues may be performed to maximise the benefit of conflation. For example, the storage device queue may be drained of messages, with conflation policies being applied to each message, and then the transmission process is conducted to transmit the messages to the clients.

The number of messages that may be conflated is dependent upon the arrival rate of messages for each topic and the transmission rate of messages to clients. The longer messages remain in the queue to which conflation is applied, the greater the opportunity there is for conflation. For example, if the transmission rate to a particular client is low messages may remain in the client queue for longer, resulting in more conflation and hence a greater reduction in messages sent to that client.

The transmission rate of messages to the client may be limited by the network or server capacity, or may be set by the system. For example, it may be desirable limit the number of updates received by a client, or to limit the bandwidth utilised to transmit updates to clients. By limiting the transmission rates conflation of the messages may be increased improving the efficiency of delivery of messages.

Figure 3 shows a schematic diagram of a client queue structure that may be utilised. Each client queue comprises a main queue 40 of messages which are held in sequence. A mailbox 41, 42, 43 is provided for each topic to which the client subscribes. The mailboxes comprise
a hashmap to the actual messages in the main client queue 40. The use of a mailbox for each topic allows efficient application of conflation techniques and allows conflation policies to be applied on a per-topic basis.

As messages are entered into the main client queue 40, an entry is also made in the relevant mailbox 41, 42, 43 to which the relevant conflation policy is applied. In Figure 3 conflation is turned off for Topic A and accordingly all messages are entered in the mailbox 41 for transmission to the client. Conflation is turned on for Topic B, resulting in a change to the messages to be sent to those clients. For Topic B it can be seen that according to the conflation policy, only the initial value B(ITAL) and the second delta (B2 Delta) are queued for transmission to the client. Although not shown in Figure 3 the client queue is kept in sync with the mailboxes.

The messages in the mailbox for each topic are transmitted to the client together, with the topics being transmitted in turn as shown at 44. Other transmission schedules may be utilised as is most appropriate.

Each client's queue may also comprise subqueues for messages of different priority, and conflation policies may be applied differently to each subqueue and between subqueues. For example, in a particular embodiment, there may be low, normal and high priority subqueues with conflation only being applied to the normal subqueue.

The conflation policy may also specify further details about how conflation is utilised. For example, it may only apply to messages with certain fields, or with data within certain bounds.

An exemplary method for conflating a message which may be performed by the systems described hereinbefore will be described in relation to Figure 4.

At step 50 a message is received by a multiplexer 200a, 200b for its respective client queue. At step 51 the message is added to the main client queue. The multiplexer 200a, 200b ascertains the topic of the message and identifies (step 52) the conflation policy in place for that topic. If conflation is turned off, an entry (step 53) is made in the topic's mailbox 41, 42, 43.

If conflation is turned on, the conflation policy is applied (step 54) between the new message and any existing messages on that topic, and at step 55 an appropriate entry is made in the respective mailbox 41, 42, 43 and client queue 40.

At step 56 messages identified in the mailboxes are sent to the respective client.
In the embodiments described above conflation has been applied to clients queues, but as
will be appreciated the described principles can be applied at any appropriate point in the
system. For example, conflation could be performed on the main storage system queue in
addition to, or instead of, conflation on the client queues.

Reference is now made to Figure 5, which illustrates various components of an exemplary
computing-based device which may implement the server 116 above. The computing-based
device may be implemented as any form of a computing and/or electronic device in which
embodiments of the pub/sub system may be implemented.

The computing-based device comprises one or more processors 602 which may be
microprocessors, controllers or any other suitable type of processors for processing computer
executable instructions to control the operation of the device in order to manage and control
transmission of messages in a publish/subscribe system. In some examples, for example
where a system on a chip architecture is used, the processors 602 may include one or more
fixed function blocks (also referred to as accelerators) which implement a part of the queue
serving and transmission methods in hardware (rather than software or firmware).

The computing-based device also comprises an input interface 604, arranged to receive
messages relating to a topic from the publishers, and at least one network interface 606
arranged to send and receive data messages over the communication network 114. In some
examples, the input interface 604 and network interface 606 can be integrated.

The computer executable instructions may be provided using any computer-readable media
that is accessible by the computing based device. Computer-readable media may include, for
example, computer storage media such as memory 608 and communications media.
Computer storage media, such as memory 608, includes volatile and non-volatile, removable
and non-removable media implemented in any method or technology for storage of
information such as computer readable instructions, data structures, program modules or
other data. Computer storage media includes, but is not limited to, RAM, ROM, EPROM,
EEPROM, flash memory or other memory technology, CD-ROM, digital versatile disks (DVD)
or other optical storage, magnetic cassettes, magnetic tape, magnetic disk storage or other
magnetic storage devices, or any other non-transmission medium that can be used to store
information for access by a computing device. In contrast, communication media may
embody computer readable instructions, data structures, program modules, or other data in a
modulated data signal, such as a carrier wave, or other transport mechanism. As defined
herein, computer storage media does not include communication media. Although the
computer storage media (memory 608) is shown within the computing-based device it will be
appreciated that the storage may be distributed or located remotely and accessed via a
network or other communication link (e.g. using network interface 606).

Platform software comprising an operating system 610 or any other suitable platform software
may be provided at the computing-based device to enable application software 612 to be
executed on the device. Additional software provided at the device may include message
selector logic 614 for implementing the functionality of the conflation manager 202 described
above, and output manager logic for implementing the functionality of the output manager 204
described above. The memory 608 can also provide a data store 618, which can be used to
provide storage for data used by the processors 602 when performing the queue serving and
transmission operations. This can include storing of the messages from the publishers and
storing of the virtual queues.

The term 'processor' and 'computer' are used herein to refer to any device with processing
capability such that it can execute instructions. Those skilled in the art will realize that such
processing capabilities are incorporated into many different devices and therefore the term
'computer' includes set top boxes, media players, digital radios, PCs, servers, mobile
telephones, personal digital assistants and many other devices.

Those skilled in the art will realize that storage devices utilized to store program instructions
can be distributed across a network. For example, a remote computer may store an example
of the process described as software. A local or terminal computer may access the remote
computer and download a part or all of the software to run the program. Alternatively, the
local computer may download pieces of the software as needed, or execute some software
instructions at the local terminal and some at the remote computer (or computer network).
Those skilled in the art will also realize that by utilizing conventional techniques known to
those skilled in the art that all, or a portion of the software instructions may be carried out by a
dedicated circuit, such as a DSP, programmable logic array, or the like.

Any range or device value given herein may be extended or altered without losing the effect
sought, as will be apparent to the skilled person.

It will be understood that the benefits and advantages described above may relate to one
embodiment or may relate to several embodiments. The embodiments are not limited to
those that solve any or all of the stated problems or those that have any or all of the stated
benefits and advantages.

Any reference to 'an' item refers to one or more of those items. The term 'comprising' is used
herein to mean including the method blocks or elements identified, but that such blocks or
elements do not comprise an exclusive list and a method or apparatus may contain additional blocks or elements.

The steps of the methods described herein may be carried out in any suitable order, or simultaneously where appropriate. Additionally, individual blocks may be deleted from any of the methods without departing from the spirit and scope of the subject matter described herein. Aspects of any of the examples described above may be combined with aspects of any of the other examples described to form further examples without losing the effect sought. Where elements of the figures are shown connected by arrows, it will be appreciated that these arrows show just one example flow of communications (including data and control messages) between elements. The flow between elements may be in either direction or in both directions.

It will be understood that the above description of a preferred embodiment is given by way of example only and that various modifications may be made by those skilled in the art. Although various embodiments have been described above with a certain degree of particularity, or with reference to one or more individual embodiments, those skilled in the art could make numerous alterations to the disclosed embodiments without departing from the spirit or scope of this invention.
Claims

1. A computer-implemented method to conflate a time-sensitive message prior to transmission of the time-sensitive message to a subscriber, the method comprising:

   providing a queue comprising at least one time-sensitive message to be transmitted to a client;

   receiving a further time-sensitive message for transmission to the client;

   searching the queue for a time-sensitive message related to the further time-sensitive message; and

   if a related time-sensitive message is identified in the queue, generating a conflated message from the further time-sensitive message and the identified time-sensitive message.

2. A method according to claim 1, wherein searching the queue comprises scanning each of the time-sensitive messages in the queue for time-sensitive messages related to the further time-sensitive message.

3. A method according to claim 1, wherein searching the queue comprises searching a lookup table for time-sensitive messages related to the further time-sensitive message, the lookup table comprising a list of topics associated with at least a portion of the time-sensitive messages in the queue.

4. A method according to any preceding claim, wherein the conflated message comprises at least one of: the content of the time-sensitive message; the content of the further time-sensitive message; the content of both the time-sensitive message and the further time-sensitive message; and new content based on the content of the time-sensitive message and the further time-sensitive message.

5. A method according to claim 4, wherein the content of the conflated message is based on an analysis of the content of the time-sensitive message and the further time-sensitive message.

6. A method according to any preceding claim, wherein the conflated message is inserted in the queue at one of: the position of the time-sensitive message, and the end of the queue.

7. A method according to claim 6, wherein if the conflated message is inserted at the end of the queue, the method further comprises nullifying the time-sensitive message.
8. A method according to claim 6, wherein if the conflated message is inserted at the position of the time-sensitive message, the method further comprises nullifying the further time-sensitive message.

9. A method according claim 1, wherein the conflated message is inserted in a further queue or mailbox.

10. A server to conflate a time-sensitive message prior to transmission of the time-sensitive message to a client, the server comprising:

   a memory configured to store a queue comprising a plurality of time-sensitive messages to be sent to the subscriber; and

   a processor configured to:

   receive a further time-sensitive message;

   search the queue for a time-sensitive message related to the further time-sensitive message;

   if a time-sensitive message is located, generate a conflated message from the time-sensitive message and the further time-sensitive message.

11. A server according to claim 10, wherein searching the queue comprises scanning each of the time-sensitive messages in the queue for messages related to the further time-sensitive message.

12. A server according to claim 10, wherein searching the queue comprises searching a lookup table for time-sensitive messages related to the further time-sensitive message, the lookup table comprising a list of topics associated with at least a portion of the time-sensitive messages in the queue.

13. A server according to any one of claims 10 to 122, wherein the conflated message comprises at least one of: the content of the further time-sensitive message; the content of the time-sensitive message; the content of both the time-sensitive message and the further time-sensitive message; and new content based on the content of the time-sensitive message and the further time-sensitive message.

14. A server according to claim 13, wherein the content of the conflated message is based on an analysis of the content of the time-sensitive message and the further time-sensitive message.
15. A server according to any one of claims 10 to 144, wherein the conflated message is inserted in the queue at one of: the position of the time-sensitive message; and the end of the queue.

16. A server according to claim 155, wherein if the conflated message is inserted at the end of the queue, the method further comprises nullifying the further time-sensitive message.

17. A server according to claim 155, wherein if the conflated message is inserted at the position of the time-sensitive message, the method further comprises nullifying the further time-sensitive message.

18. A server according to claim 10, wherein the conflated message is added to a further queue or mailbox.
Receive new message

Add message to main queue

Conflation on?

Entry in mailbox

Apply conflation policy

Entry in mailbox

Transmit messages in mailboxes to client
INTERNATIONAL SEARCH REPORT

A. CLASSIFICATION OF SUBJECT MATTER

INV. H04L29/08 H04L12/58

ADD.

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04L

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

Electronic database consulted during the international search (name of database and where practicable, search terms used)

EPO-Internal, WPI Data, COMPENDEX, INSPEC, IBM-TDB

C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category*</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
</table>

* 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 application or patent 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 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

A* document member of the same patent family

Further documents are listed in the continuation of Box C. See patent family annex.

Date of the actual completion of the international search

9 December 2013

Date of mailing of the international search report

13/12/2013

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2

NL - 2280 HV Rijswijk

Tel.: (+31-70) 340-2040

Fax: (+31-70) 340-3016

Authorized officer

Hes, Ronald
**DOCUMENTS CONSIDERED TO BE RELEVANT**

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td><strong>US 8 244 810 Bl</strong> (HALDAR DEBASHIS [US])</td>
<td></td>
</tr>
<tr>
<td></td>
<td>14 August 2012 (2012-08-14)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>column 1, line 56 - line 60</td>
<td></td>
</tr>
<tr>
<td></td>
<td>column 3, line 42 - line 61</td>
<td></td>
</tr>
<tr>
<td></td>
<td>column 5, line 1 - line 26</td>
<td></td>
</tr>
<tr>
<td></td>
<td>figure 4</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td><strong>US 2011/138400 Al</strong> (CHANDLER ALLAN T [AU])</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ET AL) 9 June 2011 (2011-06-09)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>paragraph [0006] - paragraph [0009]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>paragraph [0017] - paragraph [0018]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>paragraph [0024] - paragraph [0026]</td>
<td></td>
</tr>
</tbody>
</table>

*Category*
<table>
<thead>
<tr>
<th>Patent document cited in search report</th>
<th>Publication date</th>
<th>Patent family member(s)</th>
<th>Publication date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>US 2009063360 A1</td>
<td>05-03-2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WO 2009032493 A1</td>
<td>12-03-2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 2004224706 A1</td>
<td>11-11-2004</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 2011045855 A1</td>
<td>24-02-2011</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 2012106724 A1</td>
<td>03-05-2012</td>
</tr>
<tr>
<td>US 8244810 BI</td>
<td>14-08-2012</td>
<td>NONE</td>
<td></td>
</tr>
<tr>
<td>US 201138400 A1</td>
<td>09-06-2011</td>
<td>CN 102087615 A</td>
<td>08-06-2011</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JP 2011118895 A</td>
<td>16-06-2011</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KR 20110063313 A</td>
<td>10-06-2011</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 201138400 A1</td>
<td>09-06-2011</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 2012167116 A1</td>
<td>28-06-2012</td>
</tr>
</tbody>
</table>