(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) **World Intellectual Property Organization**
International Bureau

(43) **International Publication Date**
13 March 2008 (13.03.2008)

PCT

(10) **International Publication Number**
WO 2008/030082 A1

(51) **International Patent Classification:**
H04L 29/08 (2006.01)

(21) **International Application Number:**
PCT/NL2006/050220

(22) **International Filing Date:**
5 September 2006 (05.09.2006)

(25) **Filing Language:** English

(26) **Publication Language:** English

(71) **Applicant** *(for all designated States except US)*: **TELEFONAKTIEBOLAGET LM ERICSSON** [SE/SE]; Telefonvagen 30, S-164 83 Stockholm (SE).

(72) **Inventors; and**
(75) **Inventors/Applicants** *(for US only)*: **HENDRIKS, Jozef, Henricus, Petrus** [NL/NL]; Mostheuvelstraat 87, NL-5045 DP Tilburg (NL). **REITSMA, Erik, Jan** [NL/NL]; Heusdenhoutsestraat 20/22, NL-4817 WC Breda (NL). **ZWAAL, Hugo** [NL/NL]; Florisstraat 8, NL-4811 PW Breda (NL). **VERHEIJEN, Mieke** [NL/BE]; Burgemeester Wouterstraat 25, B-2600 Berchem (BE). **HUTSCHEMAEKERS, Mathias, Hubertus, Maria** [NL/NL]; Maasijk 17a, NL-5307 HM Poederoijen (NL).

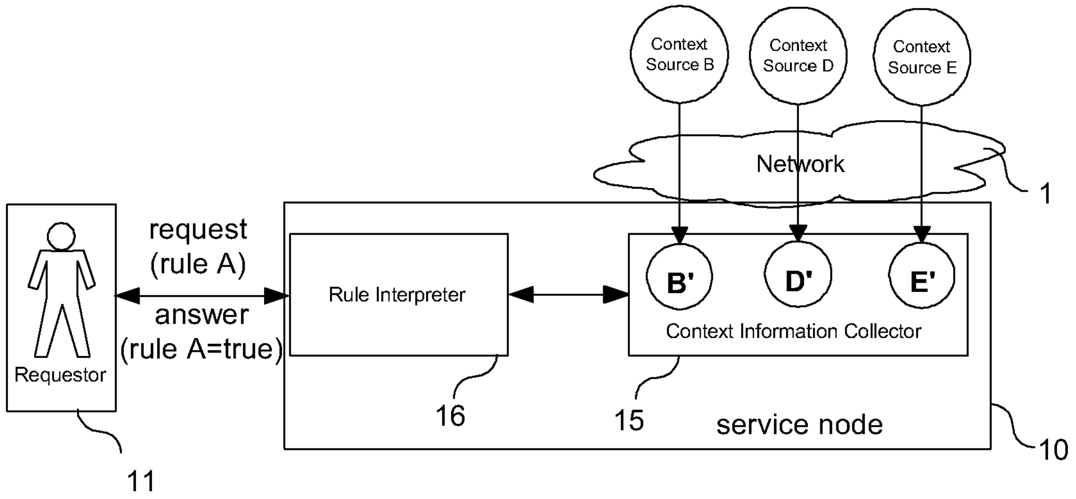
(74) **Agent: VAN WESTENBRUGGE, Andries**; Nederlandsch Octrooibureau, Postbus 29720, NL-2502 LS Den Haag (NL).

(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, 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, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(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, 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, IS, IT, LT, LU, LV, 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*

(54) **Title:** ADAPTIVE CONTEXT SOURCE UPDATE MECHANISM



(57) **Abstract:** Method for providing a service in a communication network (1), and service node (10) arranged to execute that service. The service node (10) is arranged to receive and evaluate a request from the service, the request comprising an expression having multiple elements. Each element relates to data originating from one or more context sources (B, D, E), which can provide the data to the service node (10), e.g. using context source proxies (B', D', E'). Switching between a request based mechanism and an event trigger based mechanism of one of the context sources (B, D, E) is executed based on a predetermined network parameter.

WO 2008/030082 A1

Adaptive context source update mechanism

Technical field

The present invention relates to a method for providing a service in a communication network, in which the service comprises evaluating a request comprising an expression, the expression being a function (e.g. a boolean function) of multiple elements, each element relating to data originating from one or more context sources available in the communication network. In a further aspect, the present invention relates to a service node for a communication network.

Background

Currently when a context source provides information to a requestor in a communication network, this is either done by event trigger based or request based mechanisms. Some service require the evaluation of an expression of the values of one or more context sources. For each evaluation the context sources are interrogated. This may cause an unacceptable amount of communication between the service node and the context source or context sources.

Summary

The present invention seeks to provide an improvement in network performance for a communication network providing a service involving one or more context sources.

According to the present invention, a method according to the preamble defined above is provided, in which each one of the one or more context sources updates its associated information using a request based mechanism or an event trigger based mechanism, the method comprising switching between the request based mechanism and the event trigger based mechanism of a selected one of the one or more context sources based on a predetermined network parameter. In this manner, it is possible to select an update mechanism for one or more of the context sources which is best suited given the circumstances, depending on parameters available in the communication network.

In a further embodiment, the predetermined network parameter is dependent on a first cost value related to the request based mechanism and a second cost value related to the event trigger based mechanism. The first and second cost value may be a function of one or more of the group of: frequency of occurrence of a request towards the selected context source, frequency of occurrence of an event associated with the selected context source, weight of a request, weight of an event, in which the weight of a request and the weight of an event is a function of network resources used and a monetary value. Using this embodiment, the update mechanism of a context source is adapted depending on whether it is more expensive (both in network resource usage and monetary value) to trigger on an event occurring in a context source (change of data), or to wait for an actual request for information from the respective context source.

The method may in an embodiment comprise determining the first cost value according to: $\text{Cost}_{\text{request}} = \text{Frequency}_{\text{request}} * \text{weight}_{\text{request}}$,in which $\text{Frequency}_{\text{request}}$ is the frequency of occurrence of the request, and $\text{weight}_{\text{request}}$ is a weight value representing the actual cost of a value request towards the selected context source, and in which the switching to the event trigger based mechanism is only performed when the first cost value ($\text{Cost}_{\text{request}}$) is larger than a first switching value. Again, depending on circumstances the first switching value may be selected, which allows an optimum transition to an event trigger based mechanism.

The parameter $\text{Frequency}_{\text{request}}$ is, in a further embodiment, determined based on historical data of the selected context source. This data may be available in the communication network. Also, a threshold change value may be used, in which a switch to an event trigger based mechanism is only made when $\text{Frequency}_{\text{request}}$ or the first cost value changes with an amount more than the threshold change value.

As an alternative, estimated data may be used, or also projected (future) data may be used when more suitable. E.g., in a further embodiment, $\text{Frequency}_{\text{request}}$ is determined based on an estimate by the service. Furthermore, the estimate may be further based on accuracy requirements of the data to be received from the context source. If the desired accuracy is not very high, this may result in a different switch behavior between the event trigger based mechanism and request based mechanism, which may result in more efficiency.

In a further embodiment, the first switching value is equal to the second cost value, the second cost value being determined according to $\mathrm{Cost}_{event} = \mathrm{Frequency}_{event} * \mathrm{weight}_{event}$, in which in which $\mathrm{Frequency}_{event}$ is the frequency of occurrence of the event, and weight_{event} is a weight value representing the actual cost of an event generated by the selected context source. Again, $\mathrm{Frequency}_{event}$ may be determined based on historical data of the selected context source, or alternatively, it may be based on estimated data or projected (future) data. To more accurately base a decision on switching update mechanism on precise data, in a further embodiment, a cost of switching to the trigger mechanism is added to the second cost value.

It is foreseeable that circumstances change, and that at a certain moment, the event trigger based mechanism is not longer the optimal choice for one or more of the context sources. Therefore, in a further embodiment, the switching to the query mechanism is only performed when the second cost value (Cost_{event}) is larger than a second switching value. The second switching value is in a further embodiment equal to the first cost value ($\mathrm{Cost}_{request}$).

To prevent that the switching of the update mechanism is executed too often, a hysteresis delta value is added to the first switching value, the second switching value, or both the first and second switching value in a further embodiment. This will prevent a frequent change of the update mechanism when the first and second cost value are almost equal.

In an even further embodiment, the switching is only performed for the context source of which an associated expression evaluation weight is lowest, the associated expression evaluation weight being a function of network resources used and a monetary value for evaluation of the expression towards the one or more context sources. In this manner, even more efficient use of network resources and lower cost of operation may be achieved, as the update mechanism switching and testing is only performed for a single context source.

The data of each of the plurality of context sources is in a further embodiment stored in an associated proxy server, which is directly accessible for the service. Use of a proxy server or proxy may facilitate the possibilities of making data available to applications in the network, and provide a quicker access to these data.

In a further embodiment, the method further comprises switching the proxy server to an event trigger based mechanism, in which an event is generated towards the service by the proxy server, and the associated context source is polled by the proxy server. This embodiment is useful when a context source cannot operate using an event trigger based mechanism. Towards the service, the context source proxy acts as an event driven source, with the added benefits of earlier embodiments. The polling of the context source may be executed at a predetermined frequency, again depending on circumstances as historical change data, accuracy requirements, etc.

In a further embodiment the expression is a boolean function, the data from each of the context sources being true or false. This allows a quick evaluation of the expression by the service. In an alternative embodiment, the data from the context sources is a numeric value, and the expression comprises a function of the values of the context sources. This allows easier interfacing with the context sources, as context sources may already be present or available in the network, but only provide numerical values.

The data from each of the context sources may in a further embodiment also have the value unknown. The method may further be arranged to deal with a value of unknown returned by a context source, e.g. depending on the kind of expression which is evaluated (e.g. a boolean AND or OR function). In an even further embodiment, the method further comprises to use a latest obtained value for the respective context source value if the value is unknown. In both embodiments, efficiency related to the service (e.g. the use of network resources and monetary cost) is further improved. In a further aspect, the present invention relates to a service node for a communication network providing a service in the communication network, in which the service node is arranged to receive and evaluate a request, the request comprising an expression being a function of multiple elements, each element relating to data originating from one or more context sources, the service node, in operation, being connectable to the one or more context sources, in which the service node is further arranged to execute the method according to any embodiment of the present invention. For that, the service node may further comprise a rule interpreter for evaluating the expression. Also, the service node may further comprise a context information collector for interfacing with the one or more context sources. The context information collector may in an even further embodiment comprise a context source proxy for each of the one or more context sources, for storing data associated with each respective context source.

When the context source operates according to the request based mechanism, the context source receives requests for its value from the service node and will reply to the service node with the current value. When the context source operates according to the trigger based mechanism, the context source receives a request for an event from the service node. Each time the value of the context source changes, the context source will send an event with the current value to the service node. The service node (or associated context source proxy) stores the latest received value from the context and any requests from the service for the value of the context source will not cause a request to the context source, but instead the stored latest value will be provided to the service.

With the optimizations described in this invention together the required resources for a service depending on different context sources will be reduced to the minimum without loosing the required performance in terms of latency and cost.

Brief description of the drawings

The present invention will be discussed in more detail below, using a number of exemplary embodiments, with reference to the attached drawings, in which

Fig. 1 shows a schematic diagram of a network allowing the evaluation of an expression;

Fig. 2 shows a schematic diagram of a network implementing an embodiment of the present invention;

Fig. 3 shows a schematic diagram of a network evaluating an expression according to an embodiment of the present invention; and

Fig. 4 shows a schematic diagram of a further possible scenario in a communication network using an embodiment of the present invention.

.

Detailed description

The present invention may be applied and implemented in any type of communication network structure, such as a cellular telecommunication network. The various method embodiments described below may be implemented in one of existing network hardware components or in a separate service node which interfaces with other nodes and/or components in the network 1 as required.

The basic concept of the present invention is illustrated in the network structure representation shown in Fig. 1. Only the elements needed to understand the invention are shown here. A service or application may use a network 1 to obtain the evaluation of an expression A depending on multiple elements. Actual context sources B, D, and E (providing data related to context information) may be located outside of the network 1, but are able to communicate with the network 1. E.g., the network 1 may comprise context source proxies (or context source proxy servers) B', D', and E', providing context source information in the network 1.

The network 1 shown in Fig. 1 may represent any kind of service depending on data from context sources B, D and E with certain relations or expressions A and C between them. A request from an application or service depends on the evaluation of an expression A. In the situation shown in Fig. 1, the expression A depends on two elements, i.e. another expression C (intermediate expression) and the value of context source B. The expression C depends on the value of context sources D, E as illustrated in Fig.1.

Other expressions may also be used in the embodiments of the present invention. Although the expression A illustrated above comprises one further expression (expression C), it is also envisageable that no intermediate expressions are present, i.e. the expression A to be evaluated depends directly on information (data) from one or more context sources B, D, E. Also, more complex expressions may be necessary when implementing certain network services, e.g. comprising multiple intermediate expressions.

A possible implementation of the present invention in a telecommunication network is shown in Fig. 2. Again, the context sources B, D and E are shown as being positioned outside the network 1. A requestor 11 (e.g. an application being executed in a mobile telephone operated by a user) may send a request to a service node 10, which service node 10 may be part of the network 1, or, as indicated in Fig. 2, is connected to the network 1. The service node 10 comprises a context information collector 15 connected to the context sources B, D, E, through the network 1, and a rule interpreter 16 connected to the context information collector 15 and the requestor 11. The network 1 forms the transport facility of information coming from the physical context sources

B, D, E. The physical context sources are positioned outside the service node 10, but are represented within the service node 10 by their respective proxies B', D', E'. The functionalities indicated in Fig. 2 may be implemented as software programs loaded on a processing device which is known as such, e.g. in the form of the service node 10, or another already existing processing device in the network 1. The context source proxies B', D' and E' may be implemented as part of the context information collector 15.
A request (indicated by Request(Rule A) in Fig. 2) is posted at the rule interpreter 16 by the requestor 11. In this case, the request concerns the evaluation of an expression Rule A, which is, as shown in relation to Fig. 1, based on the three context sources B, D and E. The relation between rules and context sources is already outlined in relation to Fig. 1 above. The requestor 11 is notified by the rule interpreter 16 when rule A evaluates to true. In order to have an as accurate as possible context information the context source proxies B', D', E' may be updated with an event triggering based mechanism, a request based mechanism, or a combination of both. Furthermore, the proxies may be programmed to update the information at regular intervals.

In order to have an as accurate as possible context information the context sources B, D, E will update the requestor 11 with a combination of event trigger based and request based mechanisms. This means that the context source B, D, E, or the context source proxy B', D', E' can adapt its update mechanism to the current needs.

According to an embodiment of the present invention, the mechanism used may be updated based on one or more of the following parameters or criteria, which may be available from or in the network 1:

- Frequency of occurrence of a request towards a selected context source;
- Frequency of occurrence of an event associated with a selected context source;
- Cost or weight of a request (i.e., a function of network resources used and a monetary value for interrogating a context source);
- Cost or weight of an event (i.e., a function of network resources used and a monetary value for dealing with an event from a context source);
- Relation to other context sources and whether the value is actually needed by the requestor.

The value of a context source B, D, E can be rather volatile or not. This depends heavily on the type of context source B, D, E. For instance, a calendar context source (e.g., indicating whether a person is in-meeting or not) for a person might change at the most about every hour, whereas the person's location might change every second while travelling.

The cost of a request illustrates what the actual cost is of a single value request for a particular context source B, D, E. This can be the cost for the requestor 11 in Fig. 2, or the cost for the service node 10 (more specifically the rule interpreter 16). The cost is represented by a weight value determined in terms of network resources (=N) and money =(€): weight = f $(N,€)$

The actual overall cost of a request (first cost value) for the present method embodiment is represented by the following:

$$\text{Cost}_{\text{request}} = \text{Frequency}_{\text{request}} * \text{weight}_{\text{request}}$$

in which $\text{Frequency}_{\text{request}}$ is the frequency of occurrence of the request, and $\text{weight}_{\text{request}}$ is the above defined cost of a single value request for the selected context source B, D, E. The frequency of occurrence of the request may be derived from historical data, but may also be determined as an estimate or a projected value for the future.

The actual overall cost of an event (e.g. a change of value of a context source), or a second cost value, for the present method embodiment is represented by the following:

$$\text{Cost}_{\text{event}} = \text{Cost}_{\text{triggering}} + (\text{Frequency}_{\text{event}} * \text{weight}_{\text{event}})$$

in which $\text{Frequency}_{\text{event}}$ is the frequency of occurrence of the event, and $\text{weight}_{\text{event}}$ is the above defined cost of a single event for the selected context source B, D, E. $\text{Cost}_{\text{triggering}}$ is the additional cost involved with the event trigger based mechanism. Again, the frequency of occurrence of the event may be derived from historical data, but may also be determined as an estimate or a projected value for the future.

In an alternative embodiment, the factor $\text{Cost}_{\text{triggering}}$ is neglected when determining the actual overall cost of an event.

By default triggering is turned off for each context source proxy B', D', E' in order not to consume any resources (i.e., money and network resources) before an actual request comes in (i.e. all are using a request based mechanism for updating).

In Fig. 3 it is shown that the choice between triggering and request mode (or event trigger based mechanism and request mechanism, respectively) of a context source B, D, E is influenced by the request frequency, the weight (cost) and the change (event) frequency of the actual value of the context source B, D, E.

The context source proxy B', D', E' will decide to turn into triggering mode when the first cost value ($\text{Cost}_{\text{request}}$) is larger than a first switching value, e.g. the second cost value ($\text{Cost}_{\text{event}}$).

When the context source proxy B', D', E' is in triggering mode and is triggered due to a value change, and the context source proxy did not have a pending request, the context source proxy will decide to turn off triggering when the second cost value ($\text{Cost}_{\text{event}}$) is larger than a second switching value, e.g. the first cost value ($\text{Cost}_{\text{request}}$).

This way the context source update mechanism is adapted to the fact whether it is more expensive to trigger for an event and/or wait for an actual request.

In the above, the context source proxies B', D', E' are arranged to evaluate the cost comparison, e.g. as a functionality of the context information collector 15 of the service node 10 (see Fig. 2). However, this functionality may also be provided in other parts of the system able to communicate with the context sources B, D, E, such as the context sources B, D, E, themselves, the rule interpreter 16, or the requestor 11.

Furthermore, the cost values are calculated on the basis of historical information in the above described embodiment. However, other related data may also be used, such as estimated data or projected (future) data. Also, the cost evaluation may be based on the frequency of occurrence of requests and events only, i.e. the switch between trigger mode and request mode occurs based on evaluation of the frequency of occurrence only.

In case it is known during operation of a service, that requests to a context source will occur at a certain frequency (estimate or actual knowledge), it may be more efficient to switch that context source to an event trigger based mechanism. E.g., the service may indicate to the context information collector 16 that more requests will follow in the (near) future, after which the context information collector 16 may use this information to control the context sources B, D, E, and/or the associated context source proxies B', D', E'.

When determining whether or not to switch based on the cost of the request based mechanism (first cost value $\text{Cost}_{\text{request}}$) and the cost of the event trigger based mechanism (second cost value $\text{Cost}_{\text{event})}$, it is also possible to take into account the desired accuracy of the data to be delivered by the context source. When the service indicates that no high accuracy of the data is necessary, it is then e.g. possible to switch to a mechanism with a lower load of the network 1. This may be achieved by using a trigger setting for a predetermined range of values.

In an even further embodiment, the present method embodiments as described herein may be used, even if a context source is not able to operate using an event trigger based mechanism. When during execution of the service it is determined that a context source B, D, E, should switch to an event trigger based mechanism, but cannot do this, the associated context source proxy B', D', E' is changed to a mode, in which for the service, the context source proxy acts to generate the desired events. The context source proxy B', D', E' will then poll the associated context source B, D, E, with a frequency which is based on historical data or estimated data concerning the change of value of the context source B, D, E. Again, also the desired accuracy of the data to be delivered by the context source may be taken into account, as described above.

A hysteresis mechanism will prevent a 'flickering-effect' (continuous toggling between request and trigger mode) when the $\text{Cost}_{\text{event}}$ is more or less the same as $\text{Cost}_{\text{request}}$. This can easily be implemented by adding a hysteresis delta value to the cost associated with the update mechanism not used, or by subtracting the hysteresis delta value from the cost associated with the actually used update mechanism. In the above embodiments, a hysteresis delta value may be added to the first switching value, the second switching value or both the first and second switching value.

The embodiments of the present invention as described above may be used to exploit the fact that some data will change less frequently than the data being requested for. This is shown by the following description of an example of the application of the present invention.

The expression A is evaluated for each incoming call for a subscriber (requestor 11 in Fig. 2), e.g. with a mean interval between two incoming calls of 30 minutes, i.e. two evaluations of expression A per hour. The context source B may indicate whether or not a subscriber is in a geographical location within 5 km of a centre of a town. The value of context source B is relatively constant and may e.g. change twice a day. Context source D may indicate whether or not this particular subscriber is reachable by phone (i.e. mobile phone is on, and no ongoing call is present). This value will change more frequently, e.g. about ten times per hour. Context source E may indicate whether or not the particular subscriber has any appointments at that time in his agenda (e.g. using an Outlook or Exchange program). The value of this context source E may e.g. change twice per hour.

When assuming that all the values of context sources B, D, and E are true. A naïve implementation of the evaluation of expression A will then evaluate the context sources B, D, and E twice per hour (at the request rate), which amount to 6 requests per hour. The implementation according to an embodiment of the present invention will evaluate for each context source B, D, E the number of requests and the number of (perceived) changes of the values of the context sources B, D, E. Whenever a value of a context source B, D, E changes less often than the request rate (i.e. $Cost_{event}<Cost_{request}$), the respective context source B, D, E, is switched to a trigger mode. As a result, for that context source B, D, E, an event is generated whenever the value changes, and the actual value no longer needs to be requested when evaluating expression A.

In this example, context source E changes as often as the context source E is interrogated (twice an hour). According to the present embodiment, $Cost_{event}$ and $Cost_{request}$ will be about the same, and the context source E will remain in request mode. Context source D changes value more often than the request rate (ten times per hour versus twice per hour), and thus $Cost_{event} > Cost_{request}$, according to which context source D will remain in request mode also. Context source B changes value less often than the request rate (twice a day versus twice per hour). In this case, $Cost_{event}< Cost_{request}$, and context source B will be switched to the trigger mode. The overall cost for the context sources E and D is the same for the naïve implementation and the present implementation, whereas for context source B, the present implementation will result in a lower total network cost (network load, actual cost,..). In this example, only the frequency of occurrence is accounted for. It will be clear that it is also possible to account for the weights of requests and events, and for the cost associated with switching to a trigger mode.

As indicated above, the choice for switching to the event trigger based mechanism or to the request based mechanism and vice versa may also be dependent on the relation to other context sources and whether the value is actually needed by the requestor 11. An example thereof may be to only apply the test on possible switching of update mechanisms to the context source of which the associated weight (cost) for evaluating the expression first is the lowest. As the expression to be evaluated is known for a certain application, the associated expression evaluation weights may be computed beforehand, after which only the lowest weight context source is submitted to the test for switching the update mechanism to e.g. the event trigger based mechanism.

In Fig. 4, a further illustrative example of a possible network architecture is shown for providing a service dependent on the data of more than one context source. In this example, there are two requestors 11a (Carol) and 11b (Bob), which are connected to a GPRS/UMTS network 1. The GPRS/UMTS network 1 is connected to a service node 10, which is arranged to provide an Optimal Communication Means (OCM) service, e.g. implemented as an application 17 in the service node 10. The service node 10 further comprises a Context Information Collector (CIC) 15, which is arranged to interface with a number of context sources.

Carol and Bob are subscribed to the ‘Optimal Communication Means Service’ and have defined a profile with the preferences they have upon optimal communication means in specific situations. These preferences are stored by the OCM service 17, e.g. in a database (not shown). This means that the profile of Bob is used to decide upon the list of communication means presented to Carol when he/she tries to communicate with Bob.

Carol is in her car and is trying to reach Bob who is at work in a meeting. Bob has defined in his profile that he can be reached by Instant Messaging, SMS or voice mail while in a meeting. Carol has defined in her profile that she can only use voice communication while driving a car.

Carol selects Bob from the contact list. The OCM service 17 retrieves the applicable context from both Carol and Bob through the CIC 15. The OCM service 17 discovers that because of the context of Carol and Bob there is no communication means available at this very moment and presents this through a voice message to Carol (via GPRS/UMTS network 1). The OCM service 17 also gives Carol the choice of contacting her as soon as Bob becomes available. Carol decides that she wants to contact Bob through a voice call as soon as Bob becomes available.

As soon as Bob’s meeting has ended his context changes in such a way that he is reachable through voice. The OCM service 17 decides to contact Carol and Carol can decide to call Bob. The following context sources are applicable in this scenario:
Calendar info (Bob): To determine in-meeting for instance;

Role (Carol): To determine in-car situation;

Location (Carol): To determine speed.

The various method embodiments may also be used in this example, i.e. to determine which context source is to be interrogated first based on the cost of evaluating the expression used by the OCM service 17.

The expression and intermediate expressions as used in the above embodiments and descriptions are functions of boolean elements, having a value of either true or false. In further embodiments, the elements may also comprise functions of numerical values obtained from context sources B, D, E, e.g. to check whether a value of a context source is within a certain range. In actual network implementations, some of the context sources may respond with a value 'unknown' when a value is not available, e.g. in case a cell phone is turned off, or a GPS system does not deliver any information. In most applications this will result in an exception code, which may lead to unavailability of the service.

In a further embodiment, this possibility of a context source providing a value of unknown may be utilised to further improve the efficiency of the network or to reduce cost. In a first variant, it is taken into account that next to the boolean values true and false, a context source B, D, E, may provide the value unknown as response to an interrogation. Depending on the (intermediate) expression to be evaluated, this may result in the finding that it is not necessary to interrogate any further context sources for that request, thus reducing network data traffic and interrogation cost.

In a first example, again the expressions A and C of Fig. 1 is used: intermediate expression C is true when both context sources D and E return a value true, and expression A is true when both context source B and intermediate expression C return a value true. Suppose that context source E is unavailable, then the value of intermediate expression C depends on the value of context source D. After all, when context source D returns false, intermediate expression C will evaluate to false as well, but if the value of context source D is true or unknown, the value of intermediate expression C is unknown. If the value of intermediate expression C is false, there is no need to continue asking for context source B's or E's values, which saves network resources. If the value of intermediate expression C is unknown, context source B must be interrogated.

The following table summarizes this as follows

and	E=true	E=false	E=unknown
D=true	C=true	C=false	C=unknown
D=false	C=false	C=false	**C=false**
D=unknown	C=unknown	**C=false**	C=unknown

The false cases in the unknown row and column (=bold face) are caught automatically through this invention, which is an extra optimization not taken care of before.

In the next example, the result of this embodiment will be viewed in case of a boolean OR expression. Suppose that both expression A and expression C are boolean or-expressions which return true when at least one of the context sources or child expressions is true, then the same logic applies. When context source D is unknown, then the service should still ask context source E for its value. Suppose that context source E is true, then it does not matter that context source D is unknown. After all, when one of the context sources is true, expression C results in true as well. Then, the condition to have expression C return true is satisfied, and no more network resources need to be wasted. The following table gives an overview of the return values (the case explained above can be mapped on the bold entries):

or	E=true	E=false	E=unknown
D=true	C=true	C=true	**C=true**
D=false	C=true	C=false	C=unknown
D=unknown	**C=true**	C=unknown	C=unknown

The true cases in the unknown row and column (=bold face) are caught automatically through this invention, which is an extra optimization not taken care of before.

In a further variant, the context source B, D, E, is first interrogated on availability of the respective value. Only when the context source does not return unknown will it be asked for its value, in the case of booleans, true or false, or in the case of integer range values, the value itself.

If the context source B, D, E, does return unknown, the latest value retained from a previous request can be used as input for the expression A or C.

In an example of this embodiment, for boolean values, each expression and context source contains two options, Availability and Latest. Availability consists of two choices, available and true, or available and false. When the context source value is unknown, the service can look at the latest value retained (Latest), which can also be either true or false. In this way, unknown values are no longer an issue. Unknown values are represented by the two cases in which the Availability is false. The Availability depends on the Latest and Availability values of the children elements (either expressions or context sources). Suppose expression C = A & B. Then the Availability of expression C depends on the Latest and Availability of elements A and B as follows:

Availability(C) = (Availability(A) & Availability(B))
| (Availability(A) & not Latest(A)))
| (Availability(B) & not Latest(B)))

In words, expression C is known, if elements A and B are known (in which case the value of expression C is the value of the boolean AND (A & B), or if element A is available and false (in which case the value of expression C is false) or if element B is available and false (in which case the value of expression C is false).

Initially the Latest value can be set to either true or false. As soon as the context source or expression becomes known, the correct value will be set. As long as the Availability is false, the value of Latest should not be used.

The above described embodiments and examples are not meant to limit the scope of protection of the present application, which is defined by the features as described in the appended claims.

CLAIMS

1. Method for providing a service in a communication network (1), in which the service comprises evaluating a request comprising an expression (A), the expression being a function of a plurality of elements, each element relating to data originating from one or more context sources (B, D, E) available in the communication network (1), in which each one of the one or more context sources (B, D, E) updates its associated data using a request based mechanism or an event trigger based mechanism, the method comprising switching between the request based mechanism and the event trigger based mechanism of a selected one of the one or more context sources (B, D, E) based on a predetermined network parameter.

2. Method according to claim 1, in which the predetermined network parameter is dependent on a first cost value related to the request based mechanism and a second cost value related to the event trigger based mechanism, the first and second cost value being a function of one or more of the group of: frequency of occurrence of a request towards the selected context source (B, D, E), frequency of occurrence of an event associated with the selected context source (B, D, E), weight of a request, weight of an event, in which the weight of a request and the weight of an event is a function of network resources used and a monetary value.

3. Method according to claim 2, further comprising determining the first cost value according to: $\text{Cost}_{\text{request}} = \text{Frequency}_{\text{request}} * \text{weight}_{\text{request}}$,
in which $\text{Frequency}_{\text{request}}$ is the frequency of occurrence of the request, and $\text{weight}_{\text{request}}$ is a weight value representing the actual cost of a value request towards the selected context source (B, D, E),
and in which the switching to the event trigger based mechanism is only performed when the first cost value ($\text{Cost}_{\text{request}}$) is larger than a first switching value.

4. Method according to claim 3, in which $\text{Frequency}_{\text{request}}$ is determined based on historical data of the selected context source (B, D, E).

5. Method according to claim 3, in which $\text{Frequency}_{\text{request}}$ is determined based on an estimate by the service.

6. Method according to claim 5, in which the estimate is further based on accuracy requirements of the data to be received from the context source (B, D, E).

7. Method according to any one of claims 3-6, in which the first switching value is equal to the second cost value, the second cost value being determined according to $\text{Cost}_{\text{event}} = \text{Frequency}_{\text{event}} * \text{weight}_{\text{event}}$ in which in which $\text{Frequency}_{\text{event}}$ is the frequency of occurrence of the event, and $\text{weight}_{\text{event}}$ is a weight value representing the actual cost of an event generated by the selected context source (B, D, E).

8. Method according to claim 7, in which $\text{Frequency}_{\text{event}}$ is determined based on historical data of the selected context source (B, D, E).

9. Method according to claim 7 or 8, in which a cost of switching to the trigger mechanism is added to the second cost value ($\text{Cost}_{\text{event}}$).

10. Method according to any one of claims 2-9, in which the switching to the query mechanism is only performed when the second cost value ($\text{Cost}_{\text{event}}$) is larger than a second switching value.

11. Method according to claim 10, in which the second switching value is equal to the first cost value ($\text{Cost}_{\text{request}}$).

12. Method according to claim 10 or 11, in which a hysteresis delta value is added to the first switching value, the second switching value, or both the first and second switching value.

13. Method according to any one of claims 1-12, in which the switching is only performed for the context source (B, D, E) of which an associated expression evaluation weight is lowest, the associated expression evaluation weight being a

function of network resources used and a monetary value for evaluation of the expression towards the one or more context sources (B, D, E).

14. Method according to any one of the claims 1-13, in which the data of each the plurality of context sources (B, D, E) is stored in an associated proxy server (B', D', E') accessible for the service.

15. Method according to claim 14, in which the method further comprises switching the proxy server (B', D', E') to an event trigger based mechanism, in which an event is generated towards the service by the proxy server (B', D', E'), and the associated context source ((B, D, E) is polled by the proxy server (B', D', E').

16. Method according to any one of claims 1-15, in which the expression is a boolean function, the data from each of the context sources (B, D, E) being true or false.

17. Method according to any one of claims 1-15, in which the data from the context sources (B, D, E) is a numeric value, and the expression comprises a function of the values of the context sources (B, D, E).

18. Method according to claim 16 or 17, in which the data from each of the context sources (B, D, E) may further have the value unknown.

19. Method according to claim 18, in which the method further comprises to use a latest obtained value for the respective context source value if the value is unknown.

20. Service node for a communication network providing a service in the communication network, in which the service node (10) is arranged to receive and evaluate a request, the request comprising an expression being a function of multiple elements, each element relating to data originating from one or more context sources (B, D, E), the service node (10), in operation, being connectable to the one or more context sources (B, D, E), in which the service node (10) is further arranged to execute the method according to any one of claims 1-19.

21. Service node according to claim 20, in which the service node (10) comprises a rule interpreter (16) for evaluating the expression.

22. Service node according to claim 20 or 21, in which the service node (10) comprises a context information collector (15) for interfacing with the one or more context sources (B, D, E).

23. Service node according to claim 22, in which the context information collector (15) comprises a context source proxy (B’, D’, E’) for each of the one or more context sources (B, D, E), for storing data associated with each respective context source (B, D, E).

****

Fig 1

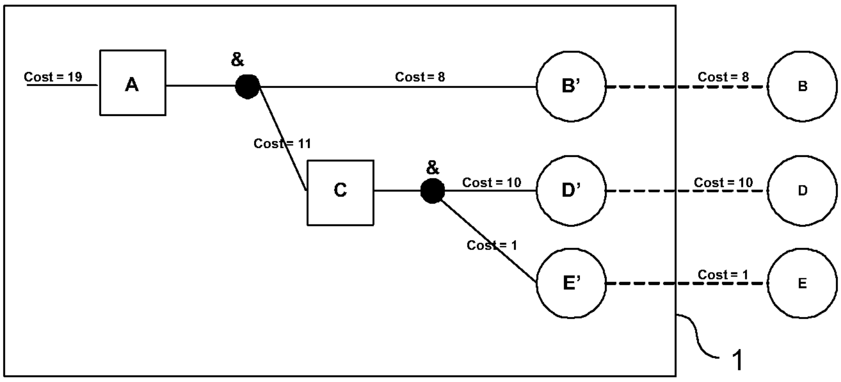


Fig 2

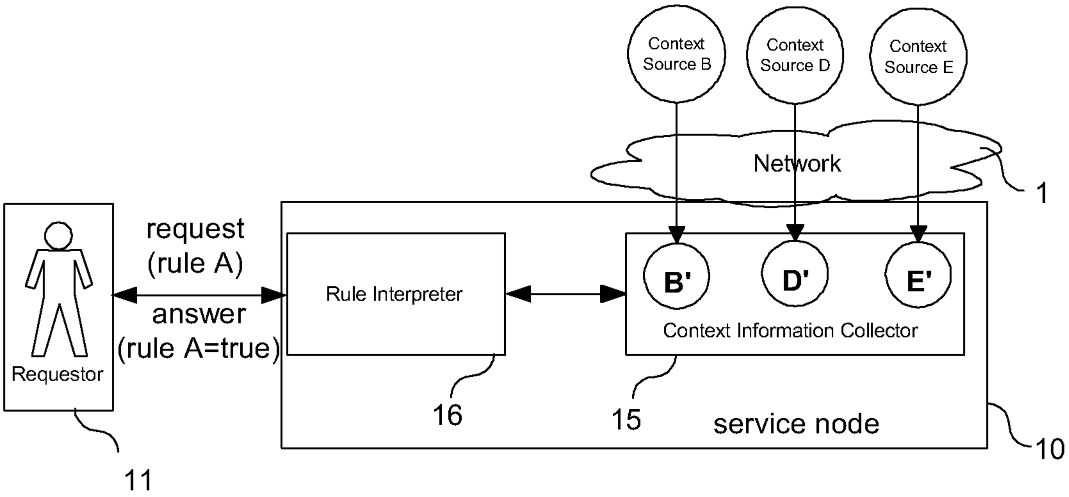


Fig 3

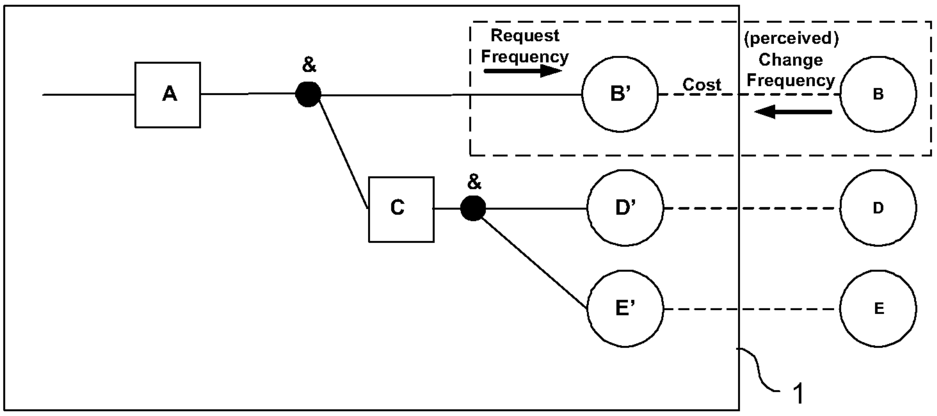
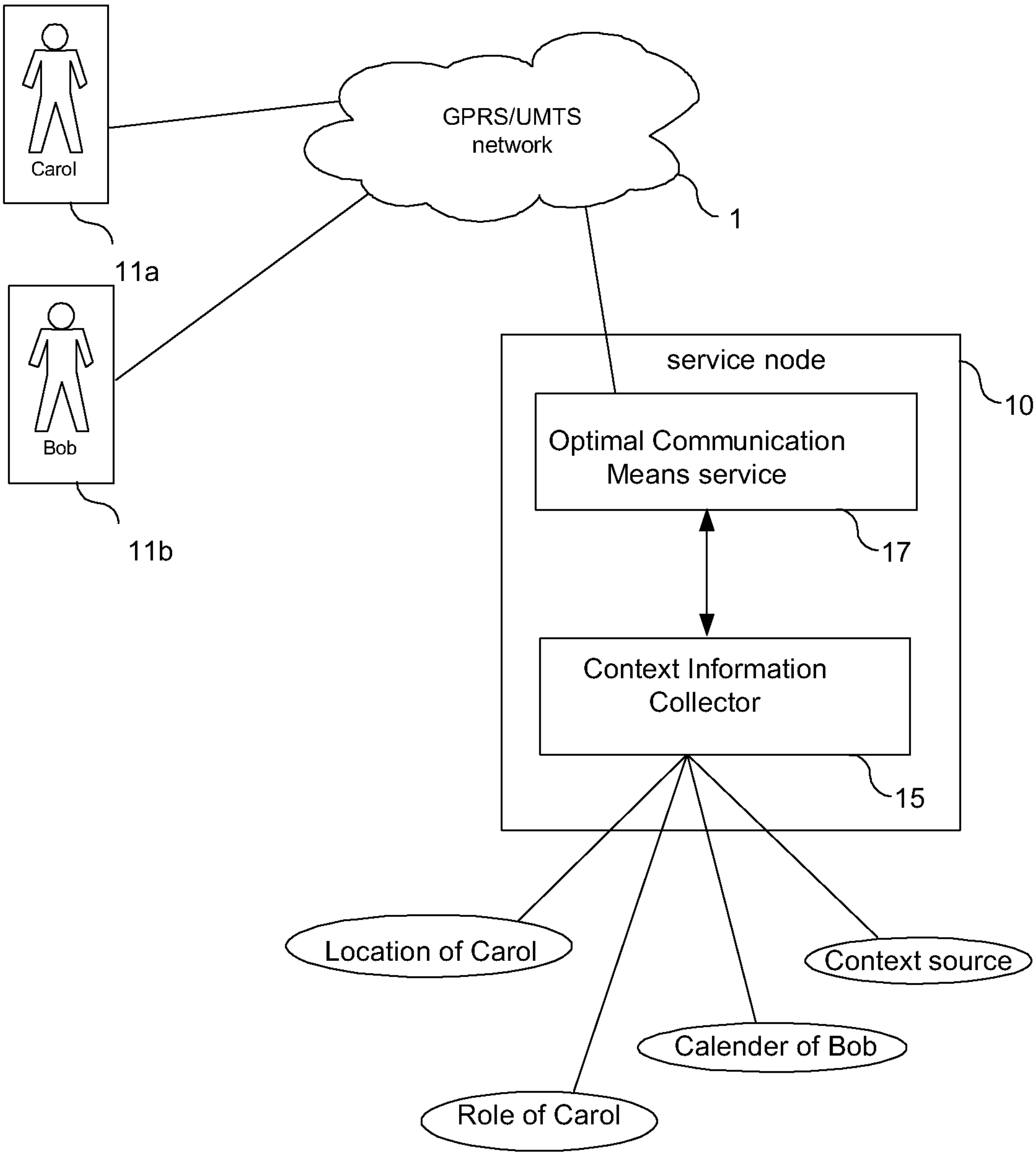


Fig 4


Carol
11a
Bob
11b
GPRS/UMTS
network
1
service node
10
Optimal Communication
Means service
17
Context Information
Collector
15
Location of Carol
Context source
Calender of Bob
Role of Carol

INTERNATIONAL SEARCH REPORT

International application No
PCT/NL2006/050220

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L29/08

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)
H04L

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, WPI Data, INSPEC, COMPENDEX

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	BHIDE M; DEOLASEE P; KATKAR A; PANCHBUDHE A; RAMAMRITHAM K; SHENOY P: "Adaptive push-pull: disseminating dynamic Web data" IEEE TRANSACTIONS ON COMPUTERS, [Online] June 2002 (2002-06), pages 652-668, XP002433530 Retrieved from the Internet: URL:http://ieeexplore.ieee.org/iel5/12/21754/01009150.pdf?tp=&arnumber=1009150&isnumber=21754> [retrieved on 2007-05-14] page 652, left-hand column, paragraph 1 - page 654, right-hand column, last paragraph page 659, left-hand column,item 4-page 660, right-hand column, item 4.4 ----- -/--	1-23

☒ Further documents are listed in the continuation of Box C.

☒ See patent family 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 14 May 2007	Date of mailing of the international search report 30/05/2007
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 Cankaya, Sukru

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	YAU S S ET AL: "An adaptive, lightweight and energy-efficient context discovery protocol for ubiquitous computing environments" DISTRIBUTED COMPUTING SYSTEMS, 2004. PROCEEDINGS. FTDCS 2004. 10TH IEEE INTERNATIONAL WORKSHOP ON FUTURE TRENDS OF SUZHOU, CHINA 26-28 MAY 2004, PISCATAWAY, NJ, USA,IEEE, 26 May 2004 (2004-05-26), pages 261-267, XP010710893 ISBN: 0-7695-2118-5 abstract page 261, item 1 page 262-page 263, item 3 page 264, items 4 and 4.1 -----	1-23
A	US 2004/039786 A1 (HORVITZ ERIC J [US] ET AL) 26 February 2004 (2004-02-26) paragraph [0147] - paragraph [0177] -----	1,20

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/NL2006/050220

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2004039786 A1	26-02-2004	US 2004098462 A1	20-05-2004