PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04Q 3/00

A2

(11) International Publication Number: **WO 98/04090**

(43) International Publication Date: 29 January 1998 (29.01.98)

(21) International Application Number: PCT/EP97/03790

(22) International Filing Date: 16 July 1997 (16.07.97)

(30) Priority Data:
9615154.3 19 July 1996 (19.07.96) GB

(71) Applicant *(for all designated States except US)*: TELEFONAKTIEBOLAGET LM ERICSSON [SE/SE]; S-126 25 Stockholm (SE).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: JELLEMA, Bart [NL/NL]; Schoorvexen 47, NL-5121 NJ Rijen (NL). PEEREN, Rene [NL/NL]; Leeuwenstraat 25, NL-4901 HA Osterhart (NL). CROUGHAN, Louise [NL/NL]; Leeuwenstraat 25, NL-4901 HA Osterhart (NL). ABEN, Freek [NL/NL]; Hoogveld 1g, NL-5447 AG Rykevoort (NL).

(74) Agent: O'CONNELL, David, Christopher; Haseltine Lake & Co., Imperial House, 15-19 Kingsway, London WC2B 6UD (GB).

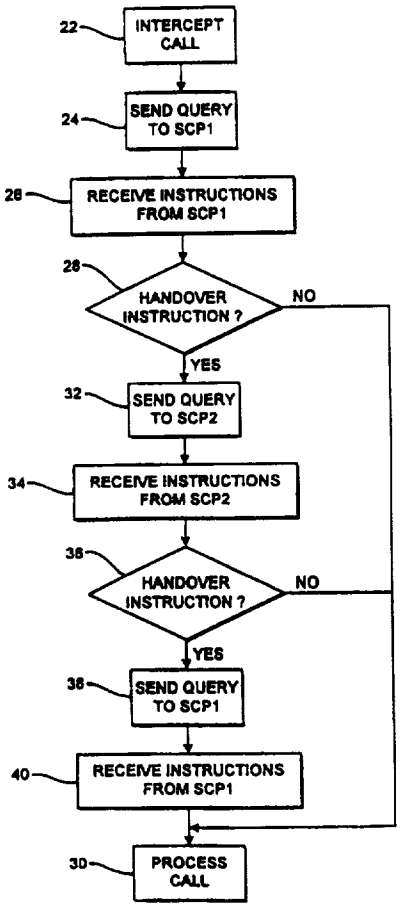
(81) Designated States: AU, BR, CA, CN, JP, KR, MX, US, European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
Without international search report and to be republished upon receipt of that report.

(54) Title: METHOD FOR HANDLING TELEPHONE CALLS

(57) Abstract

There is disclosed a method for handling telephone calls, which allows a control point to determine whether a part of a query, received from a switching point, would be better handled by a second control point. In response to such a determination, the control point returns a specific instruction to the switching point, directing it to send a query to the second control point.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

METHOD FOR HANDLING TELEPHONE CALLS

This invention relates to a method for handling telephone calls, and to a control point for use in such a method. In particular, the invention relates to a method for handling telephone calls, which allows service logic to be distributed over several control points in a network architecture.

An Intelligent Networks architecture typically includes a Service Control Point (SCP), which has a large number of Service Switching Points (SSP) connected to it. Each SSP is a switching system that can intercept telephone calls, and query the SCP. The SCP contains service specific logic and data, that allows it to return instructions to the SSP on how to deal with the intercepted call.

In some situations, however, it is advantageous to provide several SCPs in the network architecture. One such possibility arises when a SSP needs to be able to obtain data from two different databases, which are accessed through respective different SCPs.

US Patent No. 4,924,510 relates to such a situation, where a SSP needs to be able to access information stored in one of two databases at two different SCPs. In response to a dialled number, the SSP queries a first SCP. If the relevant information is not stored in the database associated with the first SCP, the first SCP returns a message to the SSP, containing a special number, which is in effect a modified version of the called party number which initiated the initial query. In response to this modified number, the SSP then sends a query to the second control point, with which the second database is associated, in order to obtain the required data from the second database. However, this method has the disadvantage that the special number returned from the first control point to the switching point needs to be

analysed in exactly the same way as the originally dialled called party number. This puts an extra processing load on the SSP.

US Patent No. 5,386,467 seeks to avoid perceived disadvantages in the method disclosed in US-4,924,510, by defining a new SCP-SCP protocol. Then, when a first SCP determines that it is unable to provide enough information to establish the required connection, it directly requests that information from the second SCP by sending an information request message. The second SCP then transmits the requested information directly to the SSP. An advantage of such a method is that it requires only three message transmissions to provide the requested information. However, this proposed solution has the disadvantage that it requires a degree of compatibility between the first and second SCPs, in that they need to share an appropriate protocol.

Thus, in the prior art, there is no way of obtaining information from two different SCPs, to deal with a single call request. As a result, all of the information required by a switching point to deal with a call request must be stored at a single SCP, and there is no possibility of a distributed arrangement of service logic.

In accordance with the present invention, the first SCP is able to return a message to the SSP, containing an explicit instruction to direct a query to a specified second SCP. This has the advantage that the message which is returned requires less processing in the SSP. Moreover, the query which is sent to the second SCP can contain information which allows the second SCP, in its turn, to return control back to the first SCP. This allows a distributed arrangement of service logic.

For a better understanding of the present invention, reference will now be made, by way of

examples, to the accompanying drawings, in which:

Figure 1 is a block schematic diagram of a network architecture in accordance with the present invention; and

Figure 2 is a flow diagram illustrating a part of the operation of the network architecture in accordance with the present invention.

As shown in Figure 1, the network includes two service control points 2, 4, and a service switching point 6. Of course, it will be appreciated that, in practice, the network will include a large number of service switching points, and may well include more than two service control points, but the illustrated part of the network is sufficient for explanation of the present invention. The service switching point (SSP) 6 intercepts calls sent from an end user 8, which are intended for transfer to other users, perhaps via other switching points. In order to connect the call correctly, the SSP 6 must obtain information from a service control point (SCP). Thus, the SCP stores service specific logic and data, that allows it to return instructions to the SSP, on how to deal with an intercepted call. In prior art networks having this basic structure, there is no way of obtaining information from two different SCPs, to deal with a single call request. As a result, each SCP must store all of the necessary information, and there is no possibility of a distributed arrangement of service logic. Although the prior art discloses network architectures, in which a switching point is able to access two different databases, there are problems associated with those proposed solutions, as discussed above, and they do not enable fully distributed service logic.

In some situations, however, it can be advantageous to execute service logic in a distributed

manner. For example, it can be advantageous to provide a large number of control points, located close to their respective users, containing the service logic relating to the more common parts of a service, for example interrogating the user with respect to the requested feature, assessing the user's rights of access to the service, or gathering statistics on the outcome of the call. The more service specific part of the logic may advantageously be executed in a centralised SCP, which is queried by a large number of SSPs. This avoids the need to install and maintain the complete SCP software and data at each location, and also avoids the disadvantage that a single centralised SCP, containing all of the service logic and data, would necessarily imply large amounts of long distance signalling traffic.

Similarly, an advantage of a distributed logic architecture would be that, even if a user were temporarily located in another operator's network, he would be able to make use of the same services as if he were calling from his home.

Figure 2 is a flow diagram, illustrating the use of the present invention, and its effect on the operation of a SSP.

In step 22, the SSP intercepts a call from an end user. In response thereto, in step 24, the SSP sends a query to a first SCP, for example SCP 2 in Figure 1. In response to the query, the SSP receives instructions in step 26 from the first SCP.

In step 28, these instructions are analysed, to determine whether they include a handover instruction.

A handover instruction is sent from the SCP, if it determines, during the execution of a service, that another SCP should take over control of the call attempt. The handover instruction includes: a parameter indicating the network address of the SCP

sending the instruction; a parameter indicating the network address of the other SCP, to which control is to be transferred; and a counter, which is incremented on handover, in order to ensure that two SCPs do not hand over to each other in an infinite loop. Advantageously, the handover instruction may also include the following additional information: a parameter containing correlation information for the SCP sending the instruction; a parameter containing correlation information for the SCP to which control is to be transferred; a parameter containing information that can be used in the set up of a TCAP dialogue towards the second SCP, such as identification of the used protocol, TCAP Dialogue Portion data etc; and a parameter containing information that was used in the set up of the TCAP dialogue towards the first SCP, in order to allow control to be handed back to that SCP.

If, in step 28, it is determined that no handover instruction has been received, the call is processed in step 30 in the conventional way. If, however, it is determined in step 28 that a handover instruction has been received, the dialogue between the SSP and the first SCP is closed, and a new dialogue with a second SCP, for example the SCP 4 in Figure 1, is opened, as a query is sent to the second SCP in step 32. This new dialogue is able to use the address and other information indicated by the first SCP. In response to the query sent to the second SCP, instructions are received in step 34. As in step 28, these instructions are analysed in step 36, to determine whether they include a handover instruction. If not, the call is processed in step 30. If a handover instruction is received, indicating that control of the call should be handed back to the first SCP, an appropriate query is sent in step 38. Then, in step 40, appropriate instructions are received from the first SCP and the

procedure passes to step 30, where the call is processed.

It will be appreciated that the second SCP may determine that a third SCP should take over control of the call attempt. If so, appropriate handover instructions may be returned to the SSP, including for example the network address of the third SCP. In such a case, the SSP then directs a query to the third SCP and receives instructions therefrom. Such instructions may allow the call to be processed or may include an instruction to handover to a further SCP.

There is thus disclosed a method of call handling which allows an efficient distribution of service logic, without requiring a new SCP to SCP interface.

In particular, it should be noted that the SSP may communicate with the two SCPs using different protocols.

CLAIMS

1. A method of handling telephone calls in a network comprising at least one switching point, to which calls are directed, and a plurality of control points, to which queries are directed from the switching points,

wherein, when a first control point receives a query from a switching point, and determines that a second control point should deal with at least a part of the query, the first control point returns a message to the switching point, instructing it to direct a query to the second control point.

2. A method as claimed in claim 1, wherein the message returned from the first control point to the switching point contains information for transmission to the second control point, allowing the second control point to return a second message to the switching point, instructing it to direct a further query to the first control point.

3. A method as claimed in claim 2, wherein the information for transmission to the second control point, contained in the message returned from the first control point to the switching point, includes the network address of the first control point.

4. A method as claimed in claim 1, wherein, when the first control point determines that it should itself deal with a first part of the query, and determines that a second control point should deal with a second part of the query, the first control point returns a message to the switching point, containing instructions relating to the first part of the query, and instructing it to direct a query to the second control point.

5. A control point, for use in a network comprising at least one switching point, to which calls are directed, and a plurality of control points, to

which queries are directed from the switching points,

the control point comprising means for determining whether a second control point should deal with at least a part of a query from a switching point, and, in response to such a determination, for returning a message to the switching point, instructing it to direct a query to a second control point.

6. A control point as claimed in claim 5, wherein the message returned from the control point to the switching point contains information for transmission to the second control point, allowing the second control point to return a second message to the switching point, instructing it to direct a further query to the control point.

7. A control point as claimed in claim 6, wherein the information for transmission to the second control point, contained in the message returned from the control point to the switching point, includes the network address of the control point.

8. A control point as claimed in claim 5, wherein in response to a determination that it should itself deal with a first part of the query, and that a second control point should deal with a second part of the query, the first control point returns a message to the switching point, containing instructions relating to the first part of the query, and instructing the switching point to direct a query to the second control point.

9. A switching point, for use in a network comprising at least one switching point, to which calls are directed, and a plurality of control points, to which queries are directed from the switching point,

the switching point comprising means for analyzing instructions returned from a first control point, and recognizing an instruction to direct a query to a specific second control point identified in the

instruction.

10. A network architecture, comprising at least one switching point, and a plurality of control points, wherein:

a first control point includes logic relating to common service functions, and

a second control point includes logic relating to service specific functions, and wherein

the first control point comprises means for determining whether the second control point should deal with at least a part of a query from a switching point, and, in response to such a determination, for returning a message to the switching point, instructing it to direct a query to the second control point.

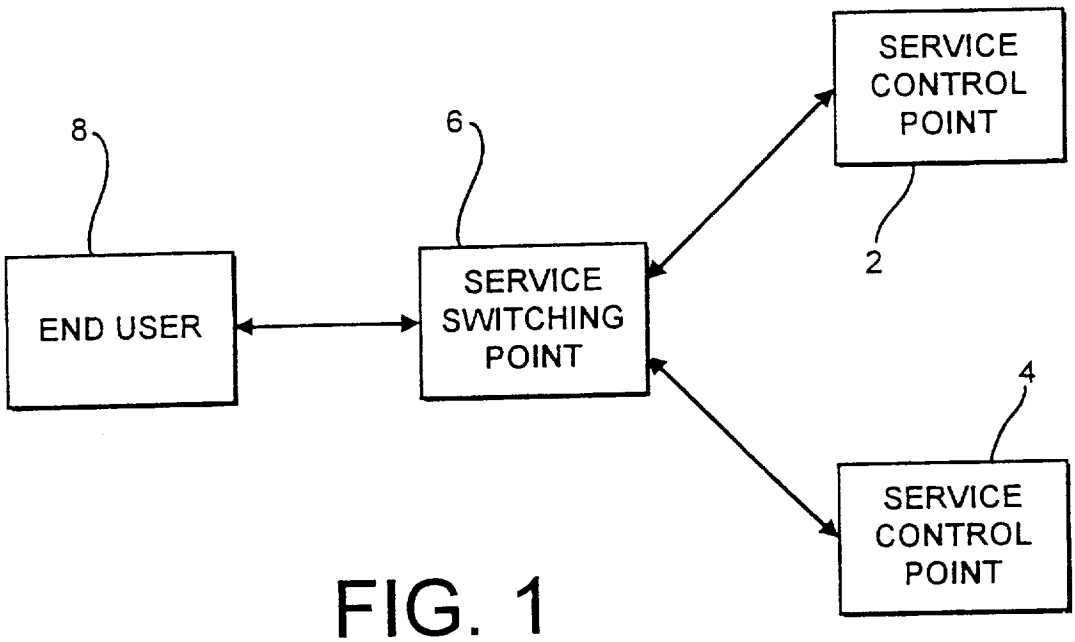

SERVICE CONTROL POINT
8
6
END USER
SERVICE SWITCHING POINT
2
4
SERVICE CONTROL POINT

FIG. 1

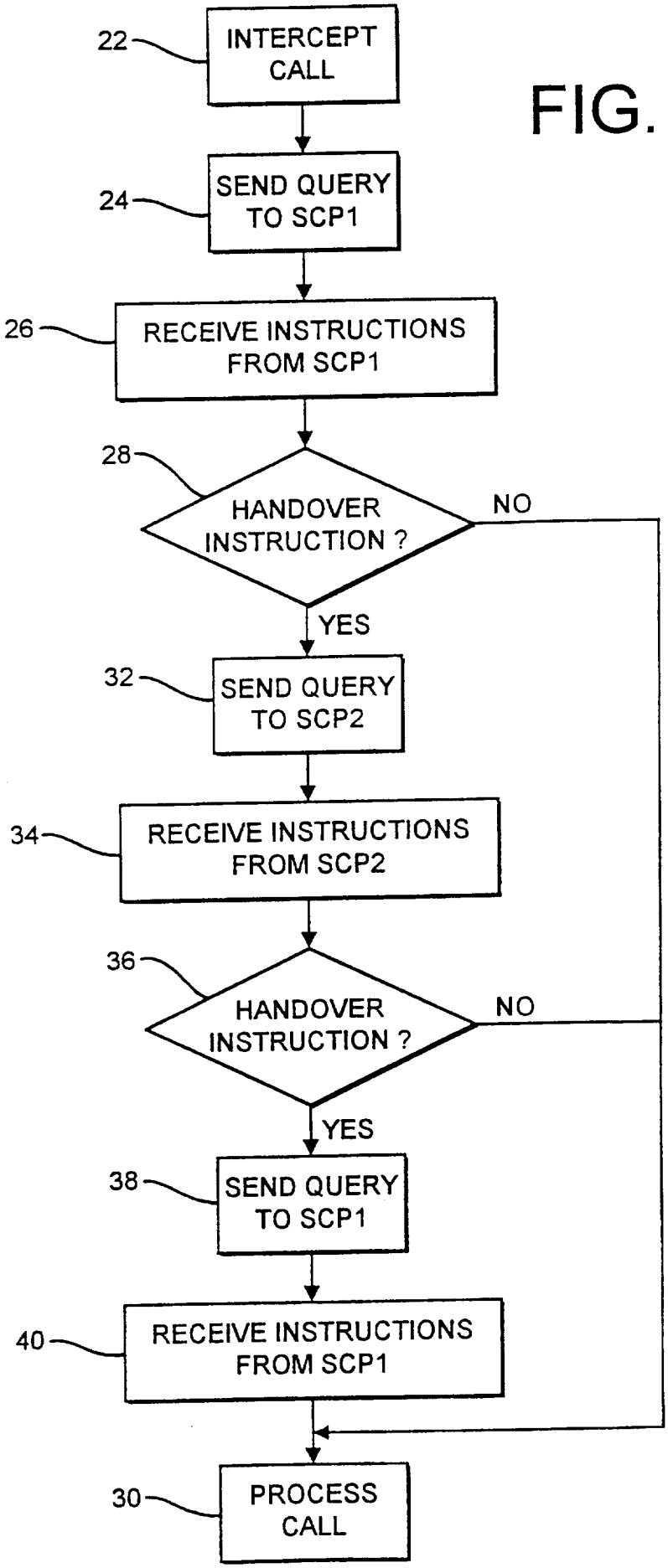

22
INTERCEPT CALL
24
SEND QUERY TO SCP1
26
RECEIVE INSTRUCTIONS FROM SCP1
28
HANDOVER INSTRUCTION ?
NO
YES
32
SEND QUERY TO SCP2
34
RECEIVE INSTRUCTIONS FROM SCP2
36
HANDOVER INSTRUCTION ?
NO
YES
38
SEND QUERY TO SCP1
40
RECEIVE INSTRUCTIONS FROM SCP1
30
PROCESS CALL

FIG. 2