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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
11 September 2009 (11.09.2009)

PCT

(10) International Publication Number
WO 2009/111146 A1

(51) International Patent Classification:
G06F 9/50 (2006.01) *H04L 29/08* (2006.01)

(21) International Application Number:
PCT/US2009/033914

(22) International Filing Date:
12 February 2009 (12.02.2009)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
12/044,778 7 March 2008 (07.03.2008) US

(71) Applicant *(for all designated States except US)*: **THE BOEING COMPANY** [US/US]; 100 North Riverside Plaza, Chicago, Illinois 60606-2016 (US).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: **SHARP, David C.** [US/US]; 12534 Ridgefield Drive, St. Louis, Missouri 63131 (US).

(74) Agents: **COUSINS, Clifford G.** et al.; The Boeing Company, P.O. Box 2515, MC 110-SD54, Seal Beach, California 90740-1515 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, 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, ST, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

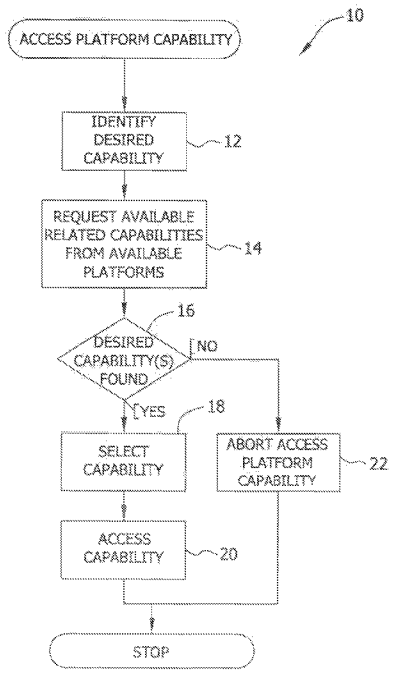
(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, TR),

[Continued on next page]

(54) Title: METHODS AND SYSTEMS FOR CAPABILITY-BASED SYSTEM COLLABORATION

(57) Abstract: A method for establishing a full end-to-end functionality chain of collaborating network accessible capabilities within a network centric environment is described. The method includes identifying the network accessible capabilities that are necessary to implement the end-to-end functionality chain, receiving data relating to which capabilities are available at each network node, determining if all necessary capabilities are available based on received responses to the data request, selecting one or more network accessible capability instances for each necessary network accessible capability, and executing the selected capability instances in an order that implements the end-to-end functionality chain.

FIG. 1



WO 2009/111146 A1

OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

METHODS AND SYSTEMS FOR CAPABILITY-BASED SYSTEM COLLABORATION

BACKGROUND

The field of the disclosure relates generally to establishing collections of collaborating networked capabilities, and more specifically, to methods and systems for capability-based system collaboration to provide end-to-end functionality that is not feasible on a single system platform.

There are many existing software systems, for example, Common Object Request Broker Architecture (CORBA), Enterprise Java Beans (EJB), and Distributed Component Object Model (DCOM), to name a few, that allow software services on different platforms to identify, discover, and collaborate with services on other platforms.

However, there are disadvantages and limitations with these existing software systems in that the range of capabilities supported by each of these systems is limited to those within a single discovery domain. In other words, configuration of collections of capabilities are limited to instantiating new capabilities and connections (e.g. CORBA Component Model (CCM) Assemblies, CORBA Trader Service, CORBA Naming Service, and CORBA Interface Repositories), and not configuring ad hoc collections of capabilities from preexisting capabilities executing independently on platforms.

In addition to these limitations, there are disadvantages and limitations with these existing software systems in that they do not provide centralized establishment and management of full end-to-end functionality chains, possibly including support for reconfiguration of the end-to-end functionality chains in the presence of faults.

Beyond these standard software systems, there is at least one known application that allows platforms to identify a single “level of interoperability” per platform and establish collaborations between platforms based on that single definition of combined available functionality.

There are disadvantages with this “level of interoperability” approach in that multiple unrelated capabilities cannot be hosted on a single platform and that

capabilities cannot be identified and incorporated into end-to-end functionality chains without first identifying platforms and platform roles. In addition, only a linear chain of capabilities culminating in a single "level of interoperability" can be identified, located, and configured per system platform.

The above described solutions do not, however, solve the problem of establishing collections of collaborating networked capabilities to provide end-to-end functionality which is infeasible on a single system platform.

BRIEF DESCRIPTION

In one aspect, a method for establishing a full end-to-end functionality chain of collaborating network accessible capabilities within a network centric environment is provided. The method includes identifying the network accessible capabilities that are necessary to implement the end-to-end functionality chain, receiving data relating to which capabilities are available at each network node, determining if all necessary capabilities are available based on received responses to the data request, selecting one or more network accessible capability instances for each necessary network accessible capability, and executing the selected capability instances in an order that implements the end-to-end functionality chain.

In another aspect, a method for providing collaborating network capabilities accessible from multiple communication frameworks within a network centric environment is provided. The method includes registering availability of each network capability with a dedicated single domain discovery service where each single domain discovery service is associated with an individual communication framework and the network capabilities accessible from the individual communication framework. The method continues with the collection of registrations from multiple single domain discovery services into a multiple domain discovery service which spans multiple communication frameworks. The method also includes retrieving, from at least one of the multiple domain discovery services, registrations for a plurality of desired network capabilities, the retrieved registrations associated with the communication frameworks the multiple domain discovery service is able to access, selecting a single registration for each desired network capability from the one or more multiple domain discovery

services, and accessing the desired network capabilities using the individual communication framework associated with the selected registration for each desired network capability.

In still another aspect, a system of computer networks is provided that includes a multiple domain discovery service and at least one processing node. The multiple domain discovery service is configured to receive registrations for collaborating network capabilities from a plurality of single domain discovery services. At least one processing node is programmed to select one or more network capabilities to be executed in an order that implements an end-to-end functionality chain and execute the selected network capabilities instances.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is an access platform capability flowchart that illustrates a method for one software application to interoperate with, and use, a capability available on a computer network.

Figure 2 is a flowchart that describes a method for configuring end-to-end functionality chains across multiple capabilities and multiple platforms.

Figure 3 is a diagram illustrating a multiple domain discovery service (MDDS) based system of systems.

DETAILED DESCRIPTION

The embodiments described herein relate to methods and associated devices for establishing collaborations among multiple networked platforms where each platform provides multiple distinct simultaneous capabilities. More specifically, the methods and device provide a capability, operable across multiple platforms, to identify and discover capabilities of interest on other networked platforms, including those platforms residing within different discovery domains. Examples of discovery domains include Common Object Request Broker Architecture (CORBA), Enterprise Java Beans (EJB), and Distributed Component Object Model (DCOM). The embodiments described herein allow for the configuration of these various discovery domain capabilities into end-to-

end network centric operation systems, due to integration of networked software across different discovery domains.

The described embodiments, in at least one aspect, focus on establishing full end-to-end functionality chains as opposed to establishing individual relationships between network capabilities. As further explained herein existing approaches are based on individual connections between executing capabilities and the defining of functionality chains prior to their execution.

Figure 1 is an access platform capability flowchart 10 that illustrates a method for one software application, referred to a "using software application" to interoperate with and use a capability available on a network. Now referring to flowchart 10, the using software application identifies 12 the desired network capability. Network capabilities are identified independently of the platform on which they execute ("host platform"), although it may be specified that some capabilities are associated in some way with a specific platform. For example, it may be specified that the capability executes on a particular platform (e.g. one with which the using software application has a high bandwidth network connection) or controls a particular platform (e.g. a specific unmanned vehicle).

In one embodiment, the using software application uses a multi-domain discovery service to request 14 all of the network capabilities which match the desired capability. Either prior to, or after, the multi-domain discovery service request, network capabilities register themselves with individual discovery services to advertise their availability within the discovery domains they support. Each discovery domain represents a set of capabilities supporting interfaces in a common language (e.g. IDL in CORBA, Java in EJB). The multi-domain discovery service supports querying for capabilities in multiple discovery domains. Results of a multi-domain discovery service query include an identifier for the capability(s) matching the desired capability(s) and the discovery domain(s) by which they may be accessed.

Capabilities belong to one or more capability categories including, but not limited to, one or more of platform independent capabilities which execute on a particular platform but otherwise have no specific platform relationship, execution platform

dependent capabilities which are constrained to execute on a particular platform(s), controlling platform dependent capabilities which execute on one platform but are statically or dynamically allocated to control and/or interface with a separate platform(s), network dependent capabilities which are constrained to execute on a particular platform(s) based on the network connectivity between that platform and some other platform (e.g. connectivity to a platform being controlled), and personnel dependent capabilities which are constrained to execute on behalf of specific system operators, possibly executing on the specific platforms with which those operators directly interact.

Characteristics of the capabilities are described in the discovery service with information referred to as capability metadata. Capability metadata includes, but is not limited to, a unique identifier of the capability itself, associated capability category(s), supported discovery domain(s), and the platform on which the capability is executing. Some capability metadata is unique to certain capabilities. Metadata for controlling platform dependent capabilities include the platform(s) being controlled and/or interfaced with. Network dependent capability metadata include expression of the relevant platforms and characteristics of network connectivity between those platforms necessary for successful operation (e.g. high bandwidth connectivity). Personnel dependent capability metadata include expressions of the personnel and/or personnel roles that must be co-resident with the capabilities desired or capabilities which must be executing on behalf of those personnel.

Referring again to the flowchart 10, the using application determines whether the desired capabilities were found 16, that is, whether any matches were returned from its multi-domain discovery service query. If multiple capabilities are returned from the query, the using application selects 18 the most desirable capability based on available capability metadata. Once a capability is selected 18 for use, the using application accesses 20 the capability using the communication services associated with the discovery domain. One example is an object request broker for CORBA. The using application may select 18 additional capabilities for use in the future if the originally selected capability fails or otherwise becomes unavailable.

If no capabilities are returned from the query, the using application aborts 22 usage of the associated platform capability. If there are one or more alternate

capabilities that may support the (potentially degraded) end-to-end functionality chain, the using application begins the process associated with flowchart 10 over again with an alternate capability.

Figure 2 is a flowchart 50 for creating a network-centric operation (NCO) system and specifically describes a method for configuring end-to-end functionality chains across multiple capabilities and multiple platforms. Referring to flowchart 50, a full set of capabilities desired for the end-to-end functionality chain is identified 52. Each capability is identified 52 as being mandatory or optional for the functionality chain. In addition, a specific set of platforms to be used for the NCO system may be identified 52. As signified by block 54 of flowchart 50, the steps described in the following paragraphs are repeated for each desired capability.

As described with respect to Figure 1, a software application uses a multi-domain discovery service to identify 56 all of the network capabilities which match the desired capability. The identified network capabilities register themselves with individual discovery services to advertise their availability within the discovery domains they support. As described previously, each discovery domain represents a set of capabilities supporting interfaces in a common language. The multi-domain discovery service supports querying for capabilities in multiple discovery domains. Results of a multi-domain discovery service query include an identifier for the capabilities matching the desired capabilities and the discovery domains by which they may be accessed.

In practical terms, a user wishes to access a capability, for example a program or other piece of software, that is running on another platform. To identify such software the user has to describe what program or other piece of software is that they are looking for, in other words, identifying 56 the entire capability. Identification 56 of capabilities, in one embodiment, includes sending out a list of characteristics or names, for example, in character strings or enumerations.

Capabilities belong to one or more capability categories including, but not limited to one or more of, platform independent capabilities which execute on a particular platform but otherwise have no specific platform relationship, execution platform(s) dependent capabilities which are constrained to execute on a particular platform,

controlling platform dependent capabilities which execute on one platform but are statically or dynamically allocated to control and/or interface with a separate platform(s), network dependent capabilities which are constrained to execute on a particular platform(s) based on the network connectivity between that platform and some other platform (e.g. connectivity to a platform being controlled), and personnel dependent capabilities which are constrained to execute on behalf of specific system operators, possibly executing on the specific platforms with which those operators directly interact.

Again referencing flowchart 50, the using application determines whether the desired capabilities were found 58, that is, whether any matches were returned from its multi-domain discovery service query. If multiple capabilities are returned from the query, the using application selects 60 the most desirable capability based on available capability metadata.

If all capabilities in the desired set of capabilities have not been selected 60, that is have not yet been found 62, the method returns to block 54 to repeat the identification, searching, and selection process for the next capability to be selected 60. Additionally, if any mandatory capability is missing 64, the method returns to block 54 to repeat the identification, searching, and selection process for the next capability to be selected 60. If any mandatory capability cannot be found, as illustrated by 64, creation of the network-centric operations (NCO) system is aborted 66.

Once all mandatory capabilities are found 58 and selected 60, all included capabilities are configured 70 and organized into an NCO system. This includes, but is not limited to, the operation of connecting interfaces between capabilities which need to interact to support the functionality chain. Once configured 70, the NCO system is initialized and begins operating 72, which is sometimes referred to as executing the NCO system.

While operating 72, the NCO system is monitored 74 for any failures or other exceptional conditions. If network connectivity fails, one or more capabilities and/or platforms may become inaccessible. Individual capabilities may fail. If one or more capabilities become unavailable 76 for any reason, the method returns to block 54 to accommodate the change. More specifically, an identification, searching, and selection

process as described above is conducted to replace the capability that has become unavailable. If the purpose of a particular NCO system is complete, the NCO system itself or an operator may terminate 78 the NCO system, releasing any reserved capabilities and network resources for other NCO systems or other purposes.

As mentioned above, there are many existing software systems, CORBA, EJB, and DCOM to name a few, which allow software services on different platforms to identify, discover, and collaborate with services on other platforms. In contrast, the above described embodiments result in an ability to identify, provide, and use multiple capabilities per platform in several different categories and in multiple discovery domains. In addition, an ability to configure multiple preexisting capabilities executing on multiple platforms into end-to-end chains of functionality in an ad hoc manner is provided as well as an ability to express intent to use capabilities for one or more end-to-end functionality chains. In a particular embodiment, expressing a usage intent may include restricting or prohibiting use of a particular capability by other functionality chains.

The described embodiments support capabilities that are made available via multiple discovery domains, in addition to supporting identification and location of related alternate capabilities which are not related to the originally desired capability by object oriented inheritance within a single language. Utilizing the embodiments, collections of networked capabilities are established which collectively work towards a single objective where there are multiple preexisting capabilities executing on potentially multiple platforms. Further, maintenance of end-to-end functionality chains is supported by informing those capabilities or controllers for those capabilities that they are in use and are not to be terminated or otherwise obstructed.

Figure 3 is an illustration of one embodiment of a multiple domain discovery service (MDDS) based system of systems 100. MDDS system of systems 100 generally illustrates that capabilities 102, 104 and 106 may be provided in different discovery domains, and registered with various single domain discovery services (SDDSs) 110, 112, 114, 116, 118, and 120. As illustrated in Figure 3, capability registrations may be shared between discovery services within each domain. Particularly, domain 1 capability registrations are shared across SDDSs 110, 114, and

118, while domain 2 capability registrations are shared across SDDSs 112, 116, and 120. The multiple domain discovery services 130, 132, and 134 are operable to allow capability registrations from both domain 1 and domain 2 to be shared. The capability registrations from multiple domains are made available to users as illustrated by capability users 140 and 150. The capability users 140 are supported by both discovery domain services 1 and 2, while the capability users 150 are supported by both discovery domain services 2 and N.

Still referring to Figure 3, when initialized, individual capabilities are registered with a single domain discovery service(s) (SDDSs) which is associated with the communication framework(s) that the discovery service supports. In Figure 3, each capability (102, 104, and 106) supports only a single communication framework, but via the configuration of the system of systems 100 approach of Figure 3, these capabilities associated with diverse communication frameworks are integrated into a single system of systems 100. These capabilities may be executing on the same or different platforms from each other and from the associated discovery services. This is true of both SDDSs or MDDSs.

The MDDSs (130, 132, 134) associated with each domain share registrations. This may be done, for example, when registrations are initially collected or when registrations are specifically requested by capability users. Each MDDS (130, 132, 134) obtains registrations from SDDSs (110, 112, 114, 116, 118, and 120) associated with multiple, and possibly all of the communication frameworks. These registrations may be obtained originally in a number of ways. In one embodiment, the MDDS may include proxies for the associated SDDSs, thereby appearing to the SDDSs as if they are standard SDDSs and collecting any registrations shared between SDDSs.

Capability users (140 and 150) which support (any set of) multiple communication frameworks query an MDDS for references to capabilities they desire. Once registrations are received, capability users assess which communication framework(s) are supported by those received registrations, select the preferred communication framework to be used for interactions, and transform the registration into the native form associated with that communication framework.

If any form of undesired response is observed when using the capability in the selected communication framework (e.g. failure to communicate, delays in communication), the capability user (140 and 150) may select an alternate registration and communication framework to access the used capability.

The provided capabilities (102, 104, and 106) in Figure 3 are illustrated as separated from the user capabilities (140 and 150), but they need not be distinct. For example, a single capability may be a provider (of itself) and also a user of other capabilities.

As compared to existing solutions, the described system of systems embodiments include an increased composition capability that allows a broader range of capabilities to be integrated into end-to-end functionality chains while allowing each platform to provide multiple, possibly unrelated, capabilities to the networked end-to-end functionality chain. In addition, an increase in control allows for centralized establishment of end-to-end functionality chains from potentially preexisting capabilities executing on potentially multiple platforms and improved maintenance of the end-to-end functionality chains by expressing usage interest in capabilities. Maintenance of end-to-end functionality chains is enhanced by informing those capabilities or controllers within the functionality chains that they are in use and are not to be terminated or otherwise obstructed.

The above described embodiments are applicable to the establishment of network centric systems and systems of systems. These types of systems are provided to customers in various markets, and are growing in importance as customers demand system integration at increasing scales. The various embodiments help to reduce the cost of implementation of network centric systems by allowing a greater diversity of capabilities that may be integrated. This integration of capabilities results in increasing amounts of software reuse while also allowing a smaller number of platforms to support the same end-to-end functionality chains by allowing a single platform to provide multiple capabilities.

The described embodiments allow a provider to integrate a broader set of capabilities in creating network centric systems than competitors, including capabilities

provided by other providers. Providers can field network centric systems of a given functionality set with fewer platforms due to increased numbers of capabilities being provided on each platform.

Customers of the providers benefit since fewer platforms are needed to support the same end-to-end functionality, thereby reducing their operations and support cost. Secondarily these customers benefit since a greater reuse of prior capability development investments results, potentially from many different providers.

While the disclosure has been described in terms of various specific embodiments, those skilled in the art will recognize that the disclosure can be practiced with modification within the spirit and scope of the claims.

WHAT IS CLAIMED IS:

1. A method for establishing a full end-to-end functionality chain of collaborating network accessible capabilities within a network centric environment, said method comprising:

identifying the network accessible capabilities that are necessary to implement the end-to-end functionality chain;

receiving data relating to which capabilities are available at each network node;

determining if all necessary capabilities are available based on received responses to the data request;

selecting one or more network accessible capability instances for each necessary network accessible capability; and

executing the selected capability instances in an order that implements the end-to-end functionality chain.

2. A method according to Claim 1 wherein executing the selected capability instances comprises causing selected network accessible capability instances to collaborate via a middleware communication framework.

3. A method according to Claim 1 wherein executing the selected capability instances comprises sharing information between the selected network accessible capabilities to establish references between the network accessible capabilities.

4. A method according to Claim 1 wherein identifying the network accessible capabilities comprises one of:

identifying the network accessible capabilities while the network centric environment is operational; and

identifying the network accessible capabilities prior to the network centric environment being operational.

5. A method according to Claim 1 wherein selecting one or more network accessible capability instances comprises assigning network accessible capability instances to one or more capability categories, where the capability categories include at least one of:

platform independent capabilities which execute on a particular platform, but have no specific platform relationship;

platform dependent capabilities which are constrained to execute on at least one particular platform;

platform dependent capabilities which execute on one platform but are statically or dynamically allocated to control and interface with a separate platform;

network dependent capabilities that are constrained to execute on a particular platform based on a network connectivity between that particular platform and another platform; and

personnel dependent capabilities that are constrained to execute on behalf of specific system operators on specific platforms with which those operators interact.

6. A method according to Claim 1 wherein identifying the network accessible capabilities comprises examining metadata associated with the network accessible capabilities to determine at least one of:

a unique identifier for the network accessible capability;

at least one associated capability category for the network accessible capability;

at least one communication framework associated with the network accessible capability; and

a network node on which the network accessible capability can be executed.

7. A method according to Claim 1 wherein at least one selected network accessible capability instance is executed or is in the process of execution prior to selection of all of the network accessible capability instances necessary to support the end-to-end functionality chain.

8. A method for providing collaborating network capabilities accessible from multiple communication frameworks within a network centric environment, said method comprising:

registering availability of each network capability with a dedicated single domain discovery service where each single domain discovery service is associated with an individual communication framework and the network capabilities accessible from the individual communication framework;

collecting registrations from multiple single domain discovery services into one or more multiple domain discovery services which span multiple communication frameworks;

retrieving, from at least one of the multiple domain discovery services, registrations for a plurality of desired network capabilities, the retrieved registrations associated with the communication frameworks the multiple domain discovery service is able to access;

selecting a single registration for each desired network capability from the one or more multiple domain discovery services; and

accessing the desired network capabilities using the individual communication framework associated with the selected registration for each desired network capability.

9. A method according to Claim 8 wherein selecting a single registration for each desired network capability comprises selecting a single registration

from registrations across multiple communication frameworks for each desired network capability.

10. A method according to Claim 8 wherein the one or more multiple domain discovery services share network capability registrations that have been received from a plurality of singe-domain discovery services.

11. A method according to Claim 8 wherein multiple instances of the same capability, at least a portion of the multiple instances of the same capability accessible from different communication frameworks, are used to support continued operation of the network centric environment in a selected single registration fails.

12. A system of computer networks, said system comprising:

a multiple domain discovery service configured to receive registrations for collaborating network capabilities from a plurality of single domain discovery services; and

at least one processing node programmed to:

select one or more network capabilities to be executed in an order that implements an end-to-end functionality chain; and

execute the selected network capabilities instances.

13. A system according to Claim 12 comprising a plurality of processing nodes, said multiple domain discovery service executable on said plurality of processing nodes, the selected instances of the one or more network capabilities executable on the various said nodes, and operable to communicate with one another and to communicate with the single-domain discovery services to collect registration information from individual communication frameworks.

14. A system according to Claim 12 wherein to execute the selected capability instances said at least one processing node is programmed to share information between said multiple domain discovery services said to propagate references to the network accessible capabilities.

15. A system according to Claim 12 wherein to select one or more network capabilities to be executed said at least one processing node is programmed to:

identify the network accessible capabilities while the network centric environment is operational; and

identify the network accessible capabilities prior to the network centric environment being operational.

16. A system according to Claim 12 wherein said at least one processing node is programmed to assign network accessible capability instances to one or more capability categories, including at least one of platform independent capabilities which execute on a particular platform, but have no specific platform relationship, platform dependent capabilities which are constrained to execute on at least one particular platform, platform dependent capabilities which execute on one platform but are statically or dynamically allocated to control and interface with a separate platform, network dependent capabilities that are constrained to execute on a particular platform based on a network connectivity between that particular platform and another platform, and personnel dependent capabilities that are constrained to execute on behalf of specific system operators on specific platforms with which those operators interact.

FIG. 1

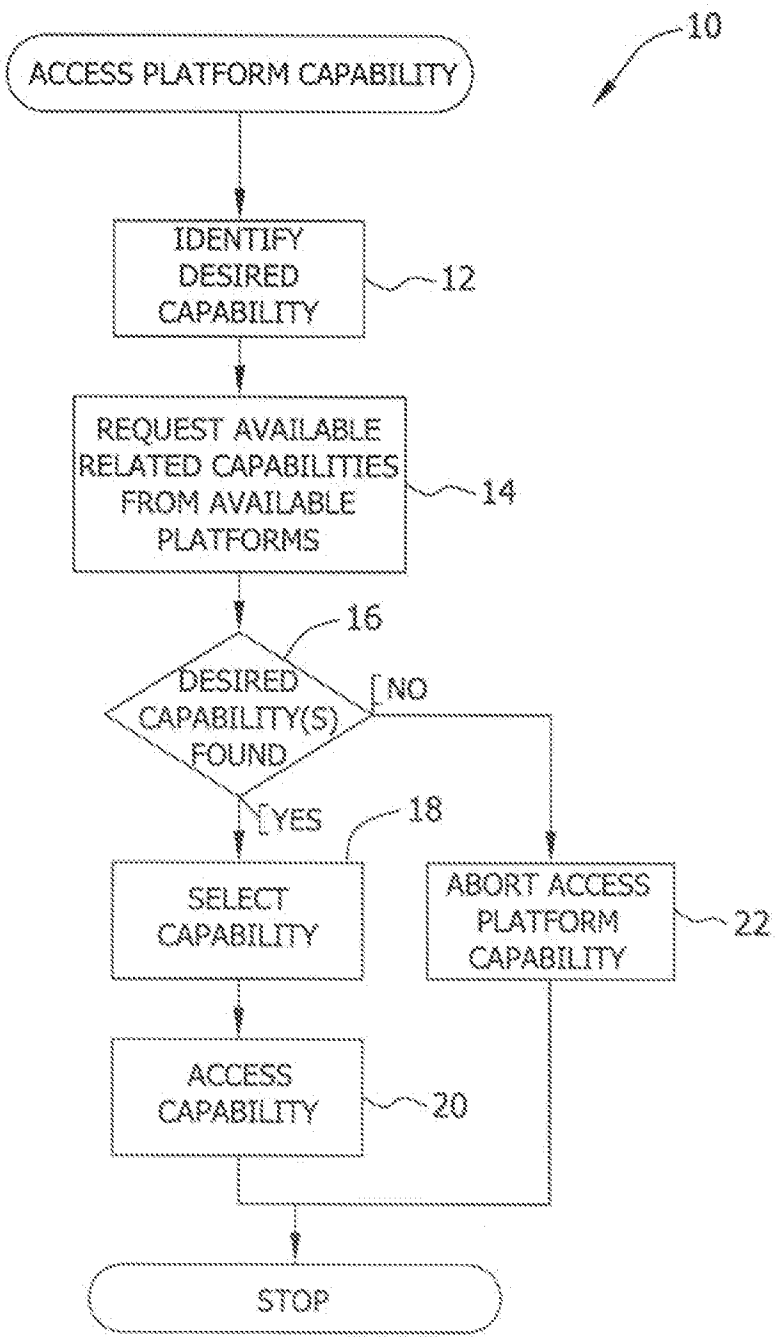

ACCESS PLATFORM CAPABILITY
10
IDENTIFY DESIRED CAPABILITY
12
REQUEST AVAILABLE RELATED CAPABILITIES FROM AVAILABLE PLATFORMS
14
16
DESIRED CAPABILITY(S) FOUND
NO
YES
18
SELECT CAPABILITY
ABORT ACCESS PLATFORM CAPABILITY
22
ACCESS CAPABILITY
20
STOP

FIG. 2

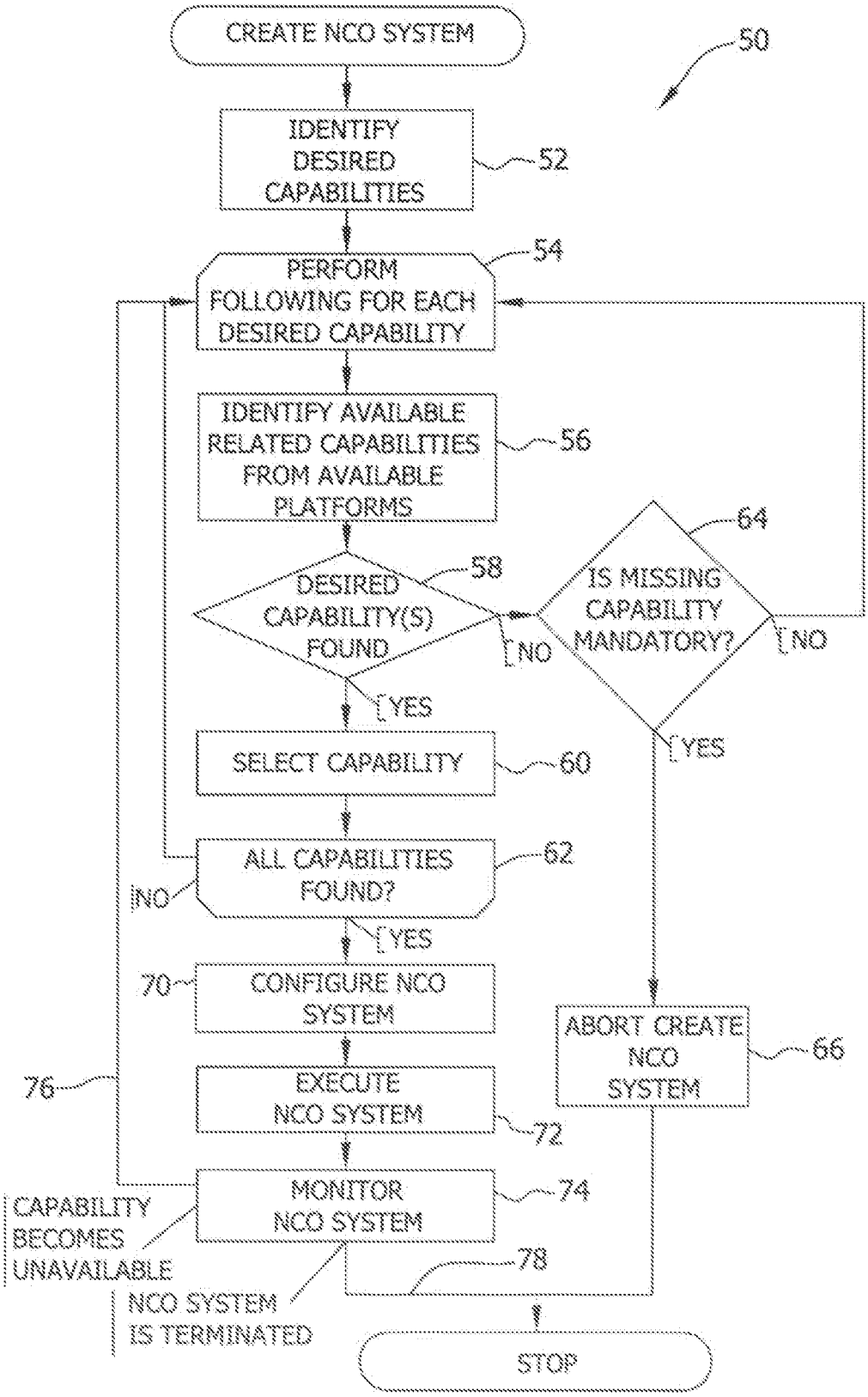

CREATE NCO SYSTEM
50
IDENTIFY DESIRED CAPABILITIES
52
PERFORM FOLLOWING FOR EACH DESIRED CAPABILITY
54
IDENTIFY AVAILABLE RELATED CAPABILITIES FROM AVAILABLE PLATFORMS
56
DESIRED CAPABILITY(S) FOUND
58
NO
YES
IS MISSING CAPABILITY MANDATORY?
64
NO
YES
SELECT CAPABILITY
60
ALL CAPABILITIES FOUND?
62
NO
YES
70
CONFIGURE NCO SYSTEM
ABORT CREATE NCO SYSTEM
66
76
EXECUTE NCO SYSTEM
72
MONITOR NCO SYSTEM
74
CAPABILITY BECOMES UNAVAILABLE
NCO SYSTEM IS TERMINATED
78
STOP

FIG. 3A

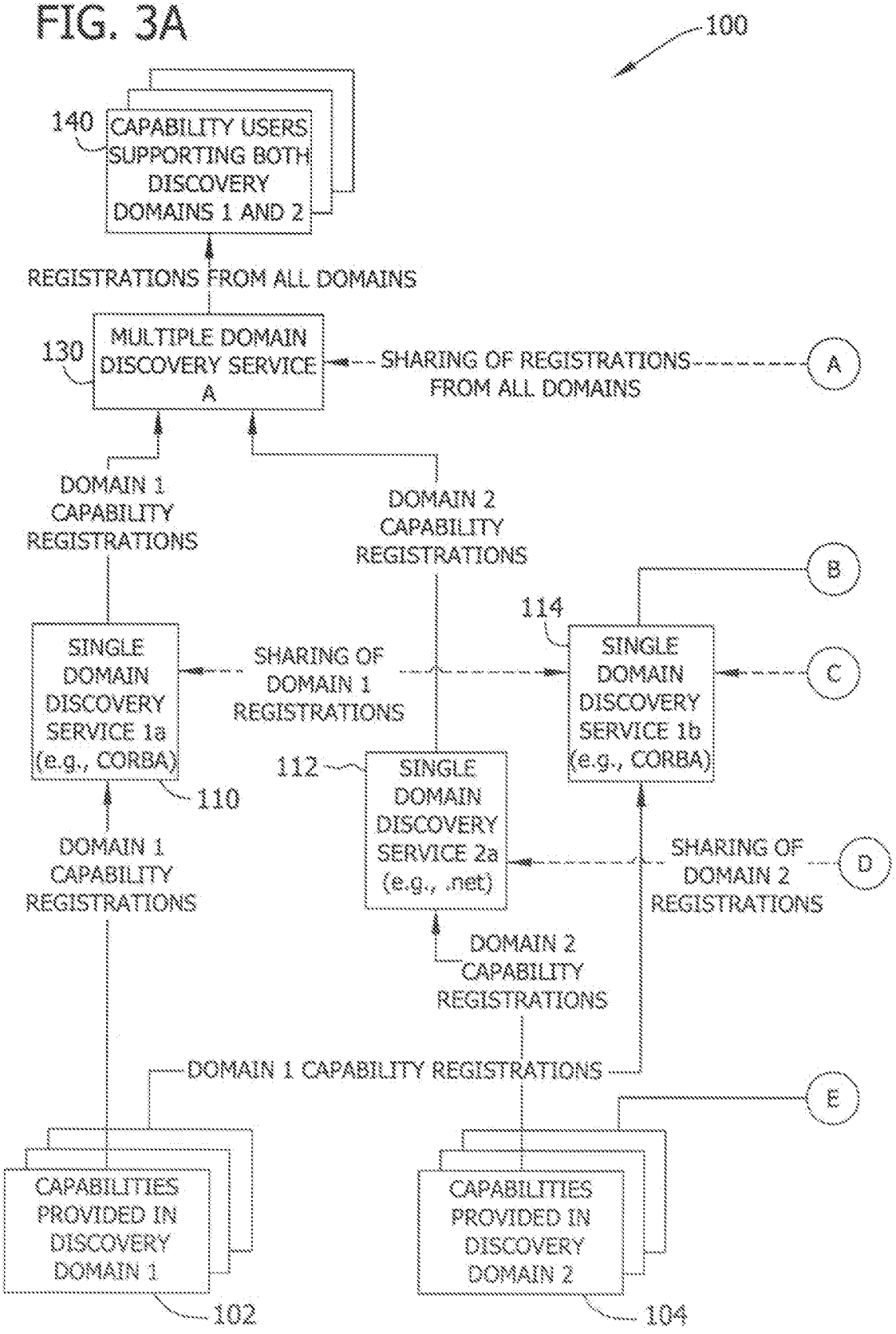
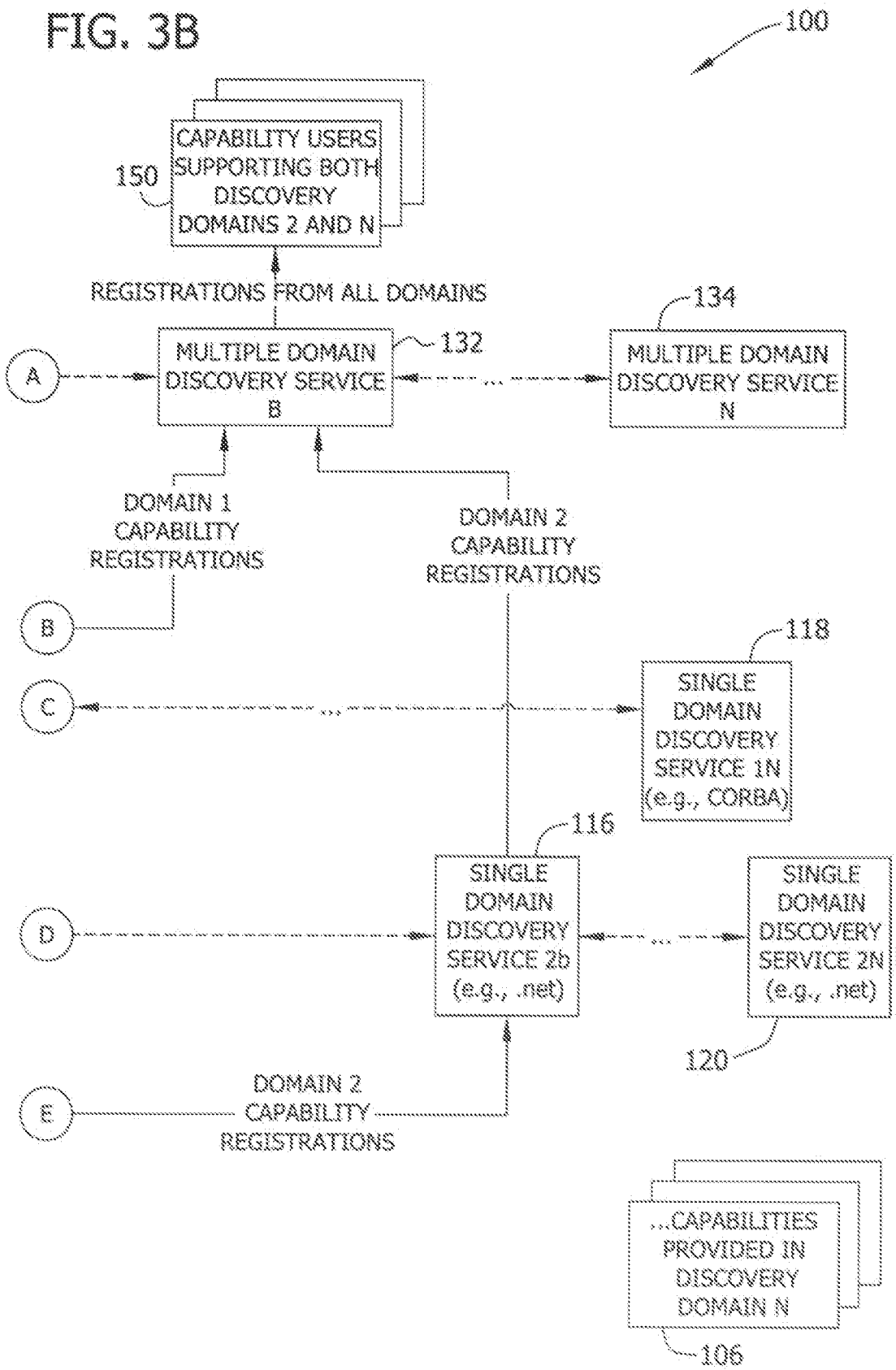

100
140
CAPABILITY USERS SUPPORTING BOTH DISCOVERY DOMAINS 1 AND 2
REGISTRATIONS FROM ALL DOMAINS
130
MULTIPLE DOMAIN DISCOVERY SERVICE A
SHARING OF REGISTRATIONS FROM ALL DOMAINS
A
DOMAIN 1 CAPABILITY REGISTRATIONS
DOMAIN 2 CAPABILITY REGISTRATIONS
B
114
SINGLE DOMAIN DISCOVERY SERVICE 1a (e.g., CORBA)
SHARING OF DOMAIN 1 REGISTRATIONS
SINGLE DOMAIN DISCOVERY SERVICE 1b (e.g., CORBA)
C
110
112
SINGLE DOMAIN DISCOVERY SERVICE 2a (e.g., .net)
DOMAIN 1 CAPABILITY REGISTRATIONS
SHARING OF DOMAIN 2 REGISTRATIONS
D
DOMAIN 2 CAPABILITY REGISTRATIONS
DOMAIN 1 CAPABILITY REGISTRATIONS
E
CAPABILITIES PROVIDED IN DISCOVERY DOMAIN 1
CAPABILITIES PROVIDED IN DISCOVERY DOMAIN 2
102
104

FIG. 3B


100
CAPABILITY USERS SUPPORTING BOTH DISCOVERY DOMAINS 2 AND N
150
REGISTRATIONS FROM ALL DOMAINS
134
MULTIPLE DOMAIN DISCOVERY SERVICE B
132
MULTIPLE DOMAIN DISCOVERY SERVICE N
A
DOMAIN 1 CAPABILITY REGISTRATIONS
DOMAIN 2 CAPABILITY REGISTRATIONS
B
118
C
SINGLE DOMAIN DISCOVERY SERVICE 1N (e.g., CORBA)
116
SINGLE DOMAIN DISCOVERY SERVICE 2b (e.g., .net)
SINGLE DOMAIN DISCOVERY SERVICE 2N (e.g., .net)
D
120
E
DOMAIN 2 CAPABILITY REGISTRATIONS
...CAPABILITIES PROVIDED IN DISCOVERY DOMAIN N
106

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2009/033914

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F9/50
ADD. 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)
G06F

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

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2007/033590 A1 (MASUOUKA RYUSUKE [US] ET AL) 8 February 2007 (2007-02-08)	1-7
A	paragraphs [0069] - [0106]	8-16
	-/--	

[X] Further documents are listed in the continuation of Box C.

[X] 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	Date of mailing of the international search report
18 May 2009	02/06/2009

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Michel, Thierry

2.

Form PCT/ISA/210 (second sheet) (April 2005)

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	LIMAM ET AL: "OSDA: Open service discovery architecture for efficient cross-domain service provisioning" COMPUTER COMMUNICATIONS, ELSEVIER SCIENCE PUBLISHERS BV, AMSTERDAM, NL, vol. 30, no. 3, 14 January 2007 (2007-01-14), pages 546-563, XP005830594 ISSN: 0140-3664 page 546, right-hand column, line 7 - line 21 page 547, left-hand column, line 36 - line 41 page 550, left-hand column, line 4 - right-hand column, line 4 page 554, right-hand column, line 1 - page 555, left-hand column, line 13 page 560, right-hand column, line 14 - page 561, left-hand column, line 4; figures 1,3	1-16
X	KOPONEN T ET AL: "A service discovery: a service broker approach" SYSTEM SCIENCES, 2004. PROCEEDINGS OF THE 37TH ANNUAL HAWAII INTERNATI ONAL CONFERENCE ON 5-8 JAN. 2004, PISCATAWAY, NJ, USA,IEEE, 5 January 2004 (2004-01-05), pages 284-290, XP010682871 ISBN: 978-0-7695-2056-8	8-12
A	page 285, right-hand column, line 35 - page 286, left-hand column, line 1 page 286, left-hand column, line 35 - line 36; figures 1-3	1-7
X	BERBNER R ET AL: "An Architecture for a QoS driven composition of Web Service based Workflows" NETWORKING AND ELECTRONIC COMMERCE RESEARCH CONFERENCE, XX, XX, 6 October 2005 (2005-10-06), XP002421684	1-7
A	the whole document	8-16
X	WO 02/102093 A (SONERA OYJ [FI]; VAENTTINEN JARI [FI]; LUUKAS PEKKA [FI]; NIEMISTOE JO) 19 December 2002 (2002-12-19)	1-7
A	the whole document	8-16

-/--

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	R. AHMED ET AL: "Service Discovery Protocols A Comparative Study" PROCEEDING OF THE IFIP/IEEE INTERNATIONAL SYMPOSIUM ON INTEGRATED NETWORK MANAGEMENT (IM'2005) APPLICATION SESSIONS, [Online] 2005, pages 1-13, XP002528555 nice, france Retrieved from the Internet: URL:http://bcr2.uwaterloo.ca/{alcatel/publ ications.htm> [retrieved on 2009-05-11] the whole document	1,5,8,12

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2009/033914

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2007033590 A1	08-02-2007	NONE	
WO 02102093 A	19-12-2002	FI 20015008 A	09-12-2002