

(12) DEMANDE INTERNATIONALE PUBLIÉE EN VERTU DU TRAITÉ DE COOPÉRATION  
EN MATIÈRE DE BREVETS (PCT)

(19) Organisation Mondiale de la Propriété  
Intellectuelle  
Bureau international



(43) Date de la publication internationale  
11 janvier 2007 (11.01.2007)

PCT

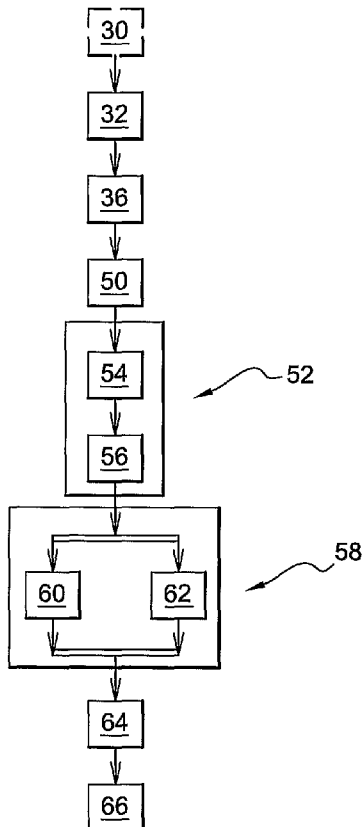
(10) Numéro de publication internationale  
**WO 2007/003774 A1**

- (51) Classification internationale des brevets :  
**G06F 17/30** (2006.01)
- (21) Numéro de la demande internationale :  
PCT/FR2006/001530
- (22) Date de dépôt international : 29 juin 2006 (29.06.2006)
- (25) Langue de dépôt : français
- (26) Langue de publication : français
- (30) Données relatives à la priorité :  
0506670 29 juin 2005 (29.06.2005) FR
- (71) Déposant (pour tous les États désignés sauf US) :  
**FRANCE TELECOM** [FR/FR]; 6 place d'Alleray,  
F-75015 Paris (FR).
- (72) Inventeur; et
- (75) Inventeur/Déposant (pour US seulement) : **BERTIN, Christian** [FR/FR]; 7 Rue Monseigneur Duchesne,  
F-35000 Rennes (FR).
- (74) Mandataires : **DE LA BIGNE, Guillaume** etc.; Cabinet  
Lhermet La Bigne & Rémy, 11 boulevard de Sébastopol,  
F-75001 Paris (FR).
- (81) États désignés (sauf indication contraire, pour tout titre de  
protection nationale disponible) : 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, MZ, NA, NG,  
NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD,

[Suite sur la page suivante]

(54) Title: METHOD OF EXCHANGING DATA BETWEEN A SERVER AND A CLIENT, SERVER, SYSTEM COMPRISING SAID SERVER, SYSTEM CLIENT, PROGRAMS FOR A SERVER-FORMING COMPUTER AND A CLIENT-FORMING COMPUTER

(54) Titre : PROCÉDE D'ÉCHANGE DE DONNÉES ENTRE UN SERVEUR ET UN CLIENT, SERVEUR CORRESPONDANT A CE PROCÉDE, SYSTÈME COMPRENANT CE SERVEUR, CLIENT DE CE SYSTÈME, PROGRAMMES CORRESPONDANT A CE PROCÉDE POUR UN ORDINATEUR FORMANT SERVEUR ET UN ORDINATEUR FORMANT CLIENT



(57) Abstract: The invention relates to a method of exchanging data between at least a first server (14) and a client (12). The inventive method comprises: a step (52) in which a request is formulated for the attention of a second server (14, 14') which, where appropriate, can be the first server, consisting in supplying structured data, particularly XML-type data; and a preliminary step (36) in which information is supplied by the first server (14) to the client (12), relating to the request (40) and response (42) structures which conform to the data structure in use on the second server (14, 14'). The information supply step (36) is preferably completed with a presentation step (50) in which said information is presented by an interface program that is executed by the client. The program generates an interface for the presentation of the information relating to the request and response structures.

(57) Abrégé : Ce procédé d'échange de données entre au moins un premier serveur (14) et un client (12) comprend une étape (52) de formulation d'une requête à l'attention d'un second serveur (14, 14') qui, le cas échéant, peut être le premier serveur, fournissant des données structurées, notamment de type XML et une étape préalable (36) de fourniture par le premier serveur (14) au client (12) d'informations relatives aux structures de requête (40) et de réponse (42) conformes à la structure de données en vigueur sur le second serveur (14, 14'). De préférence, l'étape (36) de fourniture d'informations est complétée par une étape de présentation (50) de ces informations par un programme d'interface exécuté par le client. Ce programme génère une interface de présentation des informations relatives aux structures de requête et de réponse.

WO 2007/003774 A1



SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA,  
UG, US, UZ, VC, VN, ZA, ZM, ZW.

- (84) **États désignés** (*sauf indication contraire, pour tout titre de protection régionale disponible*) : ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), eurasién (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), européen (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).

**Publiée :**

- avec rapport de recherche internationale
- avant l'expiration du délai prévu pour la modification des revendications, sera republiée si des modifications sont reçues

*En ce qui concerne les codes à deux lettres et autres abréviations, se référer aux "Notes explicatives relatives aux codes et abréviations" figurant au début de chaque numéro ordinaire de la Gazette du PCT.*

-1-

PROCEDE D'ECHANGE DE DONNEES ENTRE UN SERVEUR ET UN CLIENT, SERVEUR CORRESPONDANT A CE PROCEDE, SYSTEME COMPRENANT CE SERVEUR, CLIENT DE CE SYSTEME, PROGRAMMES CORRESPONDANT A CE PROCEDE POUR UN ORDINATEUR FORMANT SERVEUR ET UN ORDINATEUR FORMANT CLIENT

5 La présente invention concerne un procédé d'échange de données structurées entre au moins un serveur et un client, un serveur, un système comprenant ce serveur, un client de ce système, un programme pour un ordinateur formant un serveur et un programme pour un ordinateur formant le client de ce système.

10 Elle s'applique en particulier à l'exploitation automatique par le client d'informations proposées par le serveur.

Les informations se présentent sous forme de données dont la structure est généralement spécifique au serveur fournissant ces données. Ainsi un utilisateur client souhaitant accéder à ces données doit connaître la structure de données du serveur.

15 Il est connu d'échanger des données entre un client et un serveur reliés entre eux par le réseau Internet.

Ainsi, dans le cas particulier d'un serveur proposant les services d'un moteur de recherche, le client accède à une page Internet d'accueil et formule une requête en saisissant par exemple des mots clefs dans un champ de saisie de cette page d'accueil. La page d'accueil est généralement créée en langage de balisage de texte HTML (acronyme anglais de HyperText Markup Language).

20 En réponse à la requête, le moteur de recherche propose généralement une page en HTML contenant des citations comportant chacune les références d'un site Internet (dont les pages contiennent les mots clefs saisis) et un lien permettant d'accéder à ce site.

25 L'exploitation automatique des informations contenues dans la réponse fournie par le moteur de recherche peut être réalisée par le client en exécutant un programme d'analyse de cette page de réponse. Le programme d'analyse identifie des repères particuliers dans la page. Toutefois, si la présentation de la page de réponse évolue, les repères et le programme d'analyse deviennent caduques et doivent être remis à jour.

30 Pour faciliter l'obtention automatique d'informations par un client, il a été proposé dans l'état de la technique de normaliser la présentation des informations fournies par un serveur, par exemple conformément aux spécifications du type TV-Anytime ou du type à syndication vraiment simple RSS (conformément à l'acronyme anglais Really Simple Syndication).

35 Ces spécifications permettent d'organiser un échange de données structurées entre un serveur et un client en langage de balisage extensible XML (conformément à l'acronyme anglais eXtensible Markup Language).

Toutefois, la définition des spécifications de type TV-Anytime ou RSS requiert un travail de consultation des fournisseurs de services intéressés et de standardisation relativement important pour décrire la syntaxe et la sémantique des requêtes et réponses.

De plus, bien que l'accès d'un utilisateur client aux données soit simplifié, celui-ci  
5 doit se familiariser avec les spécifications précitées de la structure de données avant de pouvoir l'utiliser.

Enfin, les spécifications de type TV-Anytime ou RSS sont chacune propre à un domaine particulier, à savoir le domaine audiovisuel pour la spécification de TV-Anytime et le domaine journalistique pour la spécification RSS. Par conséquent, ces spécifications  
10 ne sont pas prévues pour être transposées à d'autres domaines.

L'invention a notamment pour but de faciliter l'échange de données structurées, notamment de type XML, entre un client et un serveur sans recourir à des travaux importants de normalisation.

A cet effet, l'invention a pour objet un procédé d'échange de données entre au  
15 moins un premier serveur et un client, le procédé étant du type comprenant une étape de formulation d'une requête à l'attention d'un second serveur qui, le cas échéant, peut être le premier serveur, fournissant des données structurées, notamment de type XML, **caractérisé en ce qu'il** comprend une étape préalable de fourniture par le premier serveur au client d'informations relatives aux structures de requête et de réponse  
20 conformes à la structure de données en vigueur sur le second serveur.

Cette étape préalable permet de rendre facilement accessible au client des données proposées par un serveur, ceci sans qu'il soit nécessaire de recourir à des systèmes de normes spécifiques à des domaines particuliers et difficiles à établir.

Le cas échéant, l'accès au serveur peut se faire automatiquement au moyen d'un  
25 programme d'accès établi à partir des structures de requête et de réponse communiquées par le serveur.

Sur Internet, ces structures sont généralement indépendantes de la présentation de la page HTML qui, habituellement, permet l'accès aux services proposés par un serveur. Une modification de la présentation de cette page est donc sans conséquence sur le  
30 fonctionnement du programme d'accès aux données.

L'utilisateur client peut donc, à partir d'un site Internet hébergé par un premier serveur, générer une requête vers un second serveur, dont le contenu est mentionné par ce site, par exemple sous forme de publicité, et sa base de données, en fonction des informations que cette base est susceptible de lui fournir.

35 Grâce à l'invention, le client est informé des structures de requête et de réponse applicables aux données de la base du serveur fournisseur de données, même si ces

structures sont spécifiques et non normées. Ceci permet au client de recueillir, éventuellement de façon automatique, des données de la base du serveur en s'affranchissant du format de la page Internet de présentation d'un service proposé par le serveur.

5 Les informations relatives à la structure de requête comprennent au moins une définition choisie parmi une définition d'un protocole de transfert de la requête, une définition d'une adresse de serveur, une définition d'une commande, et une définition d'au moins un paramètre identifié par un nom, un type et une unité, s'il est de type numérique.

10 De manière optionnelle, ces informations peuvent également comprendre :

- un groupe d'au moins deux paramètres, mis en relation à l'aide d'opérateurs logiques ;
- pour chaque paramètre, un commentaire descriptif, une liste de valeurs attendus, une valeur minimale, une valeur maximale et une valeur par défaut ;
- 15 - une liste d'éléments à inclure dans la réponse.

Par ailleurs, les informations relatives à la structure de réponse et fournies au client comprennent au moins une définition choisie parmi une définition d'un nom de structure de données attendue en réponse et une définition d'au moins un élément de la

20 réponse. identifié par un nom, un type et une unité, s'il est de type numérique.

Avantageusement, elles peuvent également comprendre, pour chaque élément, un nom, un type, une liste de valeurs possibles, un commentaire descriptif et, si l'élément est de type numérique, une unité et une cardinalité.

Cela permet en effet de pouvoir replacer à la réception de la réponse, le contenu

25 de chaque élément à l'endroit approprié, par exemple, dans une base de données client, avec ou sans conversion du codage du serveur au codage chez le client.

Selon un mode de réalisation particulier de l'invention, le premier et le second serveurs forment un seul et même serveur.

De manière optionnelle, l'étape de fourniture d'informations est complétée par une

30 étape de présentation desdites informations à l'aide d'un programme d'interface exécuté par le client .

Le programme d'interface génère, selon un mode de réalisation particulier, au moins une fenêtre présentant ces informations, par exemple sous forme de page HTML.

Le contenu de cette fenêtre est généralement rédigée dans un langage ordinaire,

35 accessible par un utilisateur qui n'est pas familier des langages informatiques.

Le programme d'interface peut en outre présenter, d'une part, des éléments descriptifs décrivant les informations relatives aux structures de requête et de réponse et, d'autre part, des éléments d'assistance à la formulation d'une requête.

5 Le cas échéant, au moins un des éléments descriptifs comprend une liste d'informations susceptibles de faire partie d'une réponse, les éléments d'assistance à la formulation d'une requête permettant de réaliser un choix parmi les informations de cette liste.

10 Cette liste peut être notamment une liste de valeurs affectées à un paramètre. Par exemple, différentes valeurs « NOM1 », « NOM2 », etc. peuvent être affectées à un paramètre « nom de société ».

L'invention évite ainsi à l'utilisateur d'avoir à formuler sa requête directement en langage XML, cette requête étant créée par choix de valeurs dans la liste proposée à l'aide d'un élément d'assistance associé au paramètre correspondant.

15 L'étape de formulation de requête peut en outre comprendre l'incorporation dans la requête de données propres au client.

Le procédé peut comprendre une étape d'émission de la requête vers le second serveur à la suite d'une action volontaire d'un utilisateur sur le client.

20 Le procédé peut également comprendre une étape d'émission automatique de la requête par le client vers le second serveur, par exemple conformément à une séquence programmée par un utilisateur dans le client.

Ainsi, une requête susceptible d'être utilisée périodiquement, par exemple une requête de cours de valeurs boursières, peut être mémorisée par le client (ce qui évite de la reformuler chaque fois) et transmise automatiquement au serveur correspondant.

25 De manière optionnelle, l'étape de présentation d'informations est initialisée par l'activation volontaire d'un lien dans une page Internet présentée par le premier serveur.

Ainsi, l'utilisateur client reçoit les informations relatives à la structure de données du serveur uniquement s'il envisage de formuler une requête à l'attention de ce serveur, ce qui évite d'utiliser les ressources du client inutilement.

30 Le procédé comprend également une étape de transmission de réponse par le second serveur, la réponse étant conforme à la structure de réponse transmise préalablement au client.

35 L'invention a également pour objet un serveur **caractérisé en ce qu'il** comprend des moyens de fourniture à un client d'informations relatives aux structures de requête et de réponse conformes à la structure de données en vigueur sur un serveur fournissant des données structurées, notamment de type XML.

Optionnellement, les deux serveurs décrits ci-dessus forment un seul et même serveur.

L'invention a également pour objet un système comprenant un serveur et un client reliés entre eux au moyen d'un réseau, le client comportant des moyens de formulation de requête à l'attention d'un second serveur qui, le cas échéant, peut être le premier serveur, fournissant des données, notamment de type XML, **caractérisé en ce que** le premier serveur est tel que défini ci-dessus.

L'invention a également pour objet un client d'un système tel que défini ci-dessus, caractérisé en ce qu'il comprend des moyens de présentation des informations fournies par le premier serveur.

Optionnellement, le client comprend des moyens d'incorporation de données qui lui sont propres dans la requête à l'attention du second serveur.

Le client peut également comprendre des moyens d'émission de la requête vers le second serveur, automatiquement ou à la suite d'une action volontaire d'un utilisateur.

De façon optionnelle, le client selon l'invention comprend également des moyens d'activation d'un lien dans une page Internet proposée par le premier serveur, déclenchant la fourniture d'informations par ce serveur.

L'invention a également pour objet un programme pour un ordinateur formant un serveur tel que défini ci-dessus, **caractérisé en ce qu'il** fournit automatiquement au client des informations relatives aux structures de requête et de réponse conformes à la structure de données d'un second serveur.

L'invention a également pour objet un programme pour un ordinateur formant un client d'un système tel que défini ci-dessus, **caractérisé en ce qu'il** génère une interface de présentation d'informations fournies au client par le serveur.

De manière optionnelle, l'interface générée par ce programme d'ordinateur:

- comprend une fenêtre, permettant d'afficher par exemple une page HTML, donnant accès ou présentant les informations fournies au client par le serveur ;
- présente, d'une part, des éléments descriptifs décrivant les informations relatives aux structures de requête et de réponse et, d'autre part, des éléments d'assistance à la formulation d'une requête.

L'invention sera mieux comprise à la lecture de la description qui va suivre, donnée uniquement à titre d'exemple et faite en se référant aux dessins dans lesquels :

- la figure 1 est un schéma d'un système d'échange de données selon un mode de réalisation particulier de l'invention ;

- la figure 2 représente les étapes successives d'un procédé d'échange de données selon l'invention, mis en œuvre dans le système de la figure 1 ;
- la figure 3 est un schéma d'une structure de requête et de réponse susceptible d'être présentée en langage XML.

5 On a représenté sur la figure 1 un système 10 selon l'invention pour l'échange de données entre un client 12 et un serveur 14 par l'intermédiaire d'un réseau classique 16 reliant le client 12 et le serveur 14, par exemple le réseau Internet.

Le client 12 comprend un ordinateur. Cet ordinateur comporte un dispositif de traitement comprenant une unité centrale 20 munie d'un processeur 20P, d'une mémoire 10 20M, par exemple de type RAM (acronyme anglais de Random Access Memory), et d'entrées-sorties 20A, 20B. L'ordinateur comporte également des éléments 22 d'interface pour un utilisateur, par exemple un clavier, un écran et une souris, et une base de données 24.

Le serveur 14 comprend un ordinateur. Cet ordinateur comporte un dispositif de 15 fourniture, comprenant une unité centrale 26, et une base de données 28. L'unité centrale 26 est munie, en général, d'un processeur 26P, d'une mémoire 26M et d'entrées-sorties 26A, 26B.

La base 24 du client 12 peut, le cas échéant, être renseignée par des données de la base 28 du serveur 14, comme précisé par la suite.

20 En variante, le client 12 et/ou le serveur 14 peuvent comprendre plusieurs bases de données. De même, le système 10 peut comprendre un serveur central donnant accès à différents serveurs fournissant des données.

On décrira ci-dessous, en référence à la figure 2, un procédé d'échange de données mis en œuvre par le système 10.

25 Dans ce qui suit, on appellera utilisateur client, un utilisateur du client 12.

Initialement (étape préliminaire 30), l'utilisateur client accède, par l'intermédiaire du réseau 16, à une page Internet proposée par le serveur 14. Cette page Internet présente des éléments formant des liens informatiques donnant accès à des données susceptibles d'être fournies par le serveur 14.

30 L'utilisateur client, intéressé par des données proposées par le serveur 14, active volontairement un lien (étape 32) donnant accès à ces données à l'aide d'un élément 22 d'interface utilisateur.

Cette étape 32 déclenche une étape 36 de fourniture, par le serveur 14, d'informations au client 12.

35 Au cours de l'étape 36, des moyens classiques (comprenant notamment l'unité centrale 26 du serveur 14) fournissent au client 12 des informations 38 comprenant des définitions

relatives à des structures de requête 40 et de réponse 42 conformes à la structure de données en vigueur sur le serveur 14. Ces moyens classiques comprennent notamment l'unité centrale 26 du serveur 14, sur lequel s'exécute un programme informatique.

5 Les informations 38 peuvent être transmises au client 12 dans un fichier de définition en langage XML programmé par un exploitant d'un service proposé par le serveur 14.

10 Les informations 38 relatives aux structures de requête 40 et de réponse 42 définissant ces structures peuvent être organisées de façon arborescente comme cela est schématisé sur la figure 3. Ainsi, la définition de la structure de requête 40 comprend au moins une définition du mode de la requête, c'est à dire du protocole de transfert de la requête. Cette dernière peut être par exemple de type 44a, conforme au protocole de transfert hypertexte HTTP (acronyme anglais de HyperText Transfer Protocol), ou de type 44b, conforme au protocole SOAP (acronyme anglais de Simple Object Access Protocol). Pour chaque protocole défini, la définition de la structure de requête 40 comprend 15 également une adresse de serveur et une commande.

La définition de la structure de requête 40 peut également comprendre une définition 46 d'au moins un paramètre d'interrogation de la base de données 28 du serveur 14, celui-ci étant identifié par un nom, un type (booléen, entier, chaîne de caractères, etc.) et une unité, s'il est de type numérique.

20 La définition de la structure de requête 40 peut également comporter une définition d'un groupe d'au moins deux paramètres, mis en relation à l'aide d'opérateurs logiques, tels que OU ou ET. Chacun des paramètres de la structure de requête 40 peut être précisé par un commentaire descriptif, une liste de valeurs attendues, une valeur minimale, une valeur maximale et une valeur par défaut. La définition de la structure de 25 requête 40 peut également comprendre une liste d'éléments à inclure dans la réponse, ces éléments étant à choisir parmi tous les éléments que peut fournir le serveur 14 dans la réponse.

30 La définition de la structure de réponse 42 peut comprendre une définition 48 d'un nom d'une structure de données attendue en réponse, et une définition d'au moins un élément de cette structure, associé éventuellement à un ou plusieurs attributs pouvant prendre des valeurs particulières.

Ainsi, dans le cas d'une base de données répertoriant des valeurs boursières de sociétés, « nom de société » et « pays du marché boursier » peuvent être des éléments. Ces éléments peuvent prendre différentes valeurs « NOM1 », « NOM2 », etc. et 35 « PAYS1 », « PAYS2 », etc.

Pour chaque élément, la définition de la structure de réponse 42 peut comprendre une définition d'un nom d'élément, d'un type (booléen, entier, chaîne de caractères, etc.), d'une liste de valeurs possibles, la signification de ces valeurs et d'une unité et d'une cardinalité, c'est-à-dire le nombre d'occurrences d'une identité, uniquement si les valeurs associées à l'élément sont numériques. Les définitions ci-dessus peuvent être

5 appréhendées par le client 12 directement à partir d'un programme capable de comprendre une structure de données en syntaxe XML selon le schéma présenté ci-dessous :

```

10 <?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="./xml_2001.xsd"/>
15  <!-- définition de l'élément principal de la structure de requête et de réponse -->
  <xs:element name="QueryResponseDescription"
    type="QueryResponseDescriptionType"/>
  <xs:complexType name="QueryResponseDescriptionType">
    <xs:sequence>
20      <xs:element name="QueryStructure" type="QueryStructureType"/>
      <xs:element name="ResponseStructure" type="ResponseStructureType"/>
    </xs:sequence>
    <xs:attribute name="version" type="xs:integer" use="required"/>
    <!-- C'est une séquence de deux éléments: structure de requête puis structure de
réponse -->
25 </xs:complexType>
  <!-- Description de la structure de requête : au choix requête HTTP ou SOAP -->
  <xs:complexType name="QueryStructureType">
    <xs:sequence>
30      <xs:choice>
        <xs:element name="HTTPQuery" type="HTTPQueryType"/>
        <xs:element name="SOAPQuery" type="SOAPQueryType"/>
      </xs:choice>
        <xs:element name="ParameterGroup" type="ParameterGoupType"
minOccurs="0"/>
35    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="HTTPQueryType">
    <xs:sequence>
40      <xs:element name="EntryPoint" type="xs:string" minOccurs="0"/>
      <xs:element name="Command" type="xs:string" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="SOAPQueryType">
    <xs:sequence>
45      <xs:element name="service" type="serviceType"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="serviceType">
    <xs:sequence>
50      <xs:element name="port" type="portType"/>

```

```

    </xs:sequence>
    <xs:attribute name="name" type="xs:string"/>
  </xs:complexType>
  <xs:complexType name="portType">
5    <xs:sequence>
      <xs:element name="SOAPAddress" type="SOAPAddressType"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string"/>
  </xs:complexType>
10 <xs:complexType name="SOAPAddressType">
    <xs:attribute name="location" type="xs:anyURI"/>
  </xs:complexType>
  <!-- Liste des paramètres de la structure de requête avec groupement possible avec
opérateurs AND et OR-->
15 <xs:complexType name="ParameterGoupType">
    <xs:sequence>
      <xs:choice maxOccurs="unbounded">
        <xs:element name="Parameter" type="ParameterType"/>
        <xs:element name="ParameterGroup" type="ParameterGoupType"/>
20      </xs:choice>
    </xs:sequence>
    <xs:attribute name="binaryBooleanOperator" default="OR">
      <xs:simpleType>
        <xs:restriction base="xs:string">
25          <xs:enumeration value="AND"/>
          <xs:enumeration value="OR"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
30 <!-- Description sémantique des paramètres de la structure de requête: texte, liste de
valeurs, valeur minimale et/ou maximale -->
  <xs:complexType name="ParameterType">
    <xs:sequence>
35      <xs:element name="Clarification" type="TextualBaseType"
maxOccurs="unbounded"/>
      <xs:element name="ValueList" type="ValueListType" minOccurs="0"/>
      <xs:element name="MinValue" type="xs:string" minOccurs="0"/>
      <xs:element name="MaxValue" type="xs:string" minOccurs="0"/>
40      <xs:element name="AllComparisonOperator" type="xs:boolean"
minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type">
45      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="xs:anyURI"/>
          <xs:enumeration value="xs:boolean"/>
          <xs:enumeration value="xs:date Time"/>
50          <xs:enumeration value="xs:float"/>
          <xs:enumeration value="xs:integer"/>
          <xs:enumeration value="xs:string"/>
        </xs:restriction>
      </xs:simpleType>

```

```

</xs:attribute>
<xs:attribute name="multipleValue" type="xs:boolean" default="false"/>
<xs:attribute name="unit" type="xs:string"/>
<xs:attribute name="use">
5   <xs:simpleType>
      <xs:restriction base="xs:string">
          <xs:enumeration value="optional"/>
          <xs:enumeration value="required"/>
      </xs:restriction>
10  </xs:simpleType>
</xs:attribute>
<!-- Nom du paramètre -->
<!-- format du paramètre -->
<!-- Possibilité de multiples valeurs ou pas pour le paramètre -->
15 </xs:complexType>
<!-- Possibilité de donner la signification du paramètre en plusieurs langues-->
<xs:complexType name="TextualBaseType">
    <xs:simpleContent>
20      <xs:extension base="xs:string">
          <xs:attribute ref="xml:lang" use="optional"/>
      </xs:extension>
    </xs:simpleContent>
</xs:complexType>
<!-- Description de la structure de réponse -->
25 <xs:complexType name="ResponseStructureType">
    <xs:choice>
        <xs:element name="SchemaRef" type="xs:anyURI"/>
        <xs:element name="Element" type="ElementType"/>
30 <!-- Description de la structure de réponse soit à partir d'un schéma connu soit
        en décrivant la structure -->
    </xs:choice>
</xs:complexType>
<!-- Description d'un élément de la réponse avec ses attributs s'il en a -->
35 <xs:complexType name="ElementType">
    <xs:sequence>
        <xs:element name="Clarification" type="TextualBaseType" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:choice>
40          <xs:element name="ValueList" type="ValueListType" minOccurs="0"/>
          <xs:element name="Element" type="ElementType" minOccurs="0"
maxOccurs="unbounded"/>
        </xs:choice>
        <xs:element name="Attribute" type="AttributeType" minOccurs="0"
maxOccurs="unbounded"/>
45 </xs:sequence>
        <xs:attribute name="name" type="xs:string" use="required"/>
        <xs:attribute name="unit" type="xs:string"/>
        <xs:attribute name="type" type="elementAttributeType"/>
        <xs:attribute name="number">
50      <xs:simpleType>
          <xs:restriction base="xs:string">
              <xs:enumeration value="0"/>
              <xs:enumeration value="1"/>
              <xs:enumeration value="unbounded"/>
          </xs:restriction>
      </xs:simpleType>
    </xs:sequence>
</xs:complexType>

```

```

        </xs:restriction>
        </xs:simpleType>
        </xs:attribute>
5      <!-- Précision de la sémantique de l'élément de réponse : nom de la balise, son
         unité, son type, sa multiplicité-->
        <!-- Possibilité d'indiquer la présence multiple de l'élément ou son absence possible
-->
        </xs:complexType>
10      <xs:complexType name="AttributeType">
        <xs:sequence>
        <xs:element name="Clarification" type="TextualBaseType" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="ValueList" type="ValueListType" minOccurs="0"/>
        </xs:sequence>
15      <xs:attribute name="name" type="xs:string" use="required"/>
        <xs:attribute name="unit" type="xs:string"/>
        <xs:attribute name="type" type="elementAttributeType"/>
        </xs:complexType>
20      <!-- Précision du type d'un élément ou d'un attribut d'élément de réponse -->
        <xs:simpleType name="elementAttributeType">
        <xs:restriction base="xs:string">
        <xs:enumeration value="xs:anyURI"/>
        <xs:enumeration value="xs:boolean"/>
        <xs:enumeration value="xs:dateTime"/>
25      <xs:enumeration value="xs:float"/>
        <xs:enumeration value="xs:integer"/>
        <xs:enumeration value="xs:string"/>
        </xs:restriction>
        </xs:simpleType>
30      <!-- Liste de valeurs possibles de l'élément ou d'un attribut d'un élément dans la
         réponse -->
        <xs:complexType name="ValueListType">
        <xs:sequence>
        <xs:element name="Value" type="ValueType" maxOccurs="unbounded"/>
35      </xs:sequence>
        </xs:complexType>
        <xs:complexType name="ValueType">
        <xs:sequence>
        <xs:element name="Clarification" type="TextualBaseType"
40      maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="content" type="xs:string" use="required"/>
        <xs:attribute name="defaultValue" type="xs:boolean"/>
        </xs:complexType>
45      </xs:schema>

```

La structure générale du fichier de définition contenant la description des structures de requêtes (40) et de réponse (42) est donc la suivante :

```

50 <QueryResponseDescription>
    <QueryStructure>
        ...
    </QueryStructure>
    <ResponseStructure>

```

```

...
</ResponseStructure>
</QueryResponseDescription>

```

5 Le procédé selon l'invention peut être réalisée par exemple selon les deux modes suivants.

Dans le premier mode de réalisation, le serveur est interrogé à l'aide du protocole de transfert HTTP.

10 Dans l'exemple de l'interrogation d'un serveur contenant des données boursières, selon ce premier mode de réalisation, la structure de requête 40 présentée dans le fichier de définition, sous la forme d'un programme en langage XML, peut être du type ci-dessous :

```

<QueryStructure>
15 <!-- requête de type HTTP avec adresse serveur et commande -->
   <HTTPQuery>
     <EntryPoint>adresse_Internet_serveur</EntryPoint>
     <Command> table XML</Command>
   </HTTPQuery>
20 <!-- Description des paramètres de la requête : nom, signification et contenu
   attendu pour chacun d'eux -->
   <ParameterGroup binaryBooleanOperator="AND">
     <Parameter name="symbole" multipleValue="true">
       <Clarification xml:lang="fr">Choisir le code de la valeur parmi la liste des
25 valeurs indiquées ici. Exemple: "CODE1" pour "SOCIETE1"</Clarification>
       <ValueList>
         <Value content="CODE2">
           <Clarification>SOCIETE2</Clarification>
30         </Value>
         <Value content=" CODE1">
           <Clarification> SOCIETE1</Clarification>
         </Value>
       </ValueList>
     </Parameter>
35 <Parameter name="language" use="optional">
       <Clarification xml:lang="fr">Langage des éléments en réponse quand
   plusieurs langues sont disponibles</Clarification>
       <ValueList>
40         <Value content="fr" defaultValue="true">
           <Clarification>Français</Clarification>
         </Value>
         <Value content="en">
           <Clarification>English</Clarification>
45         </Value>
       </ValueList>
     </Parameter>
   </ParameterGroup>
</QueryStructure>

```

Dans le second mode de réalisation, le serveur est interrogé à l'aide du protocole de transfert SOAP.

L'utilisateur client peut donc interroger le serveur à l'aide de ce protocole selon les spécifications suivantes, définissant les entrées et sorties, les codages et les liens à  
5 utiliser par les services de données:

```

<?xml version="1.0"?>
<definitions
  10   targetNamespace="urn:anyRR:transport:wSDL:2005"
      xmlns:this="urn:anyRR:transport:wSDL:2005"
      xmlns:soap="http://schemas.xmlsoap.org/wSDL/soap/"
      xmlns="http://schemas.xmlsoap.org/wSDL/"
      <documentation> Interface de Service WSDL pour toute API de fournisseur de données.
      Ce document WSDL définit les appels de l'API pour le port d'accès </documentation>
  15 <!-- Messages d'entrée et de sortie de base. -->
      <message name="get_Data">
          <part name="body" element="this:get_Data"/>
        </message>
      <message name="get_Data_Result">
  20   <part name="body" element="this:get_Data_Result"/>
        </message>
      <message name="ErrorReportMessage">
          <part name="body" element="this:ErrorReport"/>
        </message>
  25 <!-- Les différents types de services (ports) avec leurs messages d'entrée et de sortie. -->
      <portType name="get_Data_Port">
          <operation name="get_Data">
              <input message="this:get_Data"/>
              <output message="this:get_Data_Result"/>
  30   <fault name="error" message="this:ErrorReportMessage"/>
            </operation>
          </portType>
      <!-- Le "binding" définit comment SOAP/HTTP sont utilisés pour transporter le service. -->
      <binding name="get_Data_SOAP" type="this:get_Data_Port">
  35   <documentation>any data provider get_Data binding</documentation>
          <soap:binding style="document"
              transport="http://schemas.xmlsoap.org/soap/http"/>
          <operation name="get_Data">
              <soap:operation soapAction="get_Data"/>
  40   <input>
                  <soap:body use="literal" parts="body"/>
                </input>
              <output>
                  <soap:body use="literal" parts="body"/>
  45   </output>
              <fault name="error">
                  <soap:fault use="literal"/>
                </fault>
              </operation>
  50 </binding>
    </definitions>

```

La structure de requête présentée dans le fichier de définition se présente alors sous la forme suivante:

```

5  <QueryStructure>
    <!-- requête de type SOAP avec adresse serveur et commande -->
    <SOAPQuery>
        <service name="serveur">
            <port name="get_Data_serveur">
10         <SOAPAddress location="http://adresse_serveur/accesSOAP"/>
            </port>
        </service>
    </SOAPQuery>
    <!-- Description des paramètres de la requête : nom, signification et contenu
15 attendu pour chacun d'eux -->
    <ParameterGroup>
        ...
        définition des paramètres de même type que pour le premier mode de réalisation
        ...
20    </ParameterGroup>
</QueryStructure>

```

La structure de réponse 42 présentée dans le fichier de définition, sous la forme d'un programme en langage XML, ne dépend pas du protocole utilisé et peut être du type

25 ci-dessous :

```

<ResponseStructure>
    <!--Description de la structure de la réponse à la requête ci-dessus -->
    <Element name="response">
30    <Clarification xml:lang="fr">Enveloppe de la réponse</Clarification>
    <!-- Description (syntaxe et sémantique) des éléments de premier niveau de la
    structure de la réponse à la requête ci-dessus -->
    <Element name="NomValeur" type="xs:string">
        <Clarification xml:lang="fr">Nom courant de la valeur</Clarification>
35    </Element>
    <Element name="CodeValeur" type="xs:string">
        <Clarification xml:lang="fr">Code de la valeur selon la codification "xxxx".
    Exemple: "CODE1" pour "SOCIETE1"</Clarification>
    </Element>
40    <Element name="Cours" type="xs:float" unit="EUR">
        <Clarification xml:lang="fr">Cours de la valeur en Euros</Clarification>
    </Element>
    <!-- Description d'un élément de premier niveau avec plusieurs valeurs possibles
-->
45    <Element name="Conseil" type="xs:integer">
        <Clarification xml:lang="fr">Conseil des analystes</Clarification>
        <ValueList>
            <Value content="1">
                <Clarification xml:lang="fr">Vendre</Clarification>
50            </Value>
            <Value content="2">

```

-15-

```

    <Clarification xml:lang="fr">Souspondérer</Clarification>
  </Value>
  <Value content="3">
    <Clarification xml:lang="fr">Conserver</Clarification>
  </Value>
  <Value content="4">
    <Clarification xml:lang="fr">Surpondérer</Clarification>
  </Value>
  <Value content="5">
    <Clarification xml:lang="fr">Acheter</Clarification>
  </Value>
</ValueList>
</Element>
</Element>
<!-- Attribut du premier élément (response) du premier niveau -->
<Attribute name="DateHeure" type="xs:dateTime">
  <Clarification xml:lang="fr">Date et heure du cours de la valeur</Clarification>
</Attribute>
</ResponseStructure>

```

La structure de réponse 42 peut également être du type présenté ci-dessous. Elle diffère du type présenté précédemment en ce qu'elle permet de renvoyer une arborescence d'éléments, et, ainsi, dans le cadre de l'exemple, il est possible de renvoyer des informations boursières sur plusieurs sociétés dans la même réponse :

```

<ResponseStructure>
  <!-- Description de réponse. "response" est l'élément de premier niveau -->
  <Element name="response">
    <Clarification xml:lang="fr">Enveloppe de la réponse</Clarification>
    <!-- Un seul type d'élément au deuxième niveau mais il peut y avoir plusieurs
éléments de ce type dans la réponse -->
    <Element name="Valeur" number="unbounded">
      <!-- Cet élément au deuxième niveau peut apparaître plusieurs fois dans la
réponse comme indiqué par number="unbounded" ci dessus -->
      <!-- Premier élément à l'intérieur de l'élément "Valeur", donc au troisième
niveau -->
      <Element name="CodeValeur" type="xs:string" >
        <Clarification xml:lang="fr">Code de la valeur selon la codification "xxxx".
Exemple: "CODE1" pour "SOCIETE1"</Clarification>
      </Element>
      <Element name="Cours" type="xs:float" unit="EUR">
        <Clarification xml:lang="fr">Cours de clôture de la valeur en
Euros</Clarification>
      </Element>
      <Element name="Variation" type="xs:float">
        <Clarification xml:lang="fr">variation du cours par rapport au cours de
clôture précédent en pourcent</Clarification>
      </Element>
      <Element name="Lien" type="xs:anyURI">
        <Clarification xml:lang="fr">URL pour accéder à la page complète des
Echos sur la valeur courante</Clarification>
      </Element>

```

```

    </Element>
    <!-- Attribut du premier élément (response) du premier niveau -->
    <Attribute name="DateHeure" type="xs:dateTime">
      <Clarification xml:lang="fr">Date et heure du cours de la ou des valeurs qui
5 suivent</Clarification>
    </Attribute>
  </Element>
</ResponseStructure>

```

10 On comprend donc que l'étape de fourniture consiste en une fourniture par le premier serveur 14 au client 12 de structures de requête 40 et de réponse 42 conformes à la structure de données en vigueur sur le second serveur 14 ou 14', ces structures étant définies dans un programme en langage de description de données.

15 A l'aide du fichier fourni par le serveur 14, par exemple du type d'au moins un des programme ci-dessus, le client 12 peut formuler une requête adaptée aux informations qu'il veut obtenir de la part du serveur 14.

La requête dont les éléments ont été explicités par un fichier de définition selon l'exemple peut être formulée à l'aide du protocole de transfert http, de la façon suivante:

*http://adresse\_Internet\_serveur/tableXML?symbole=CODE1+CODE2&language=fr.*

20 L'utilisateur peut également formuler une requête en choisissant les éléments qu'il veut voir apparaître dans la réponse :

*http://adresse\_Internet\_serveur/tableXML?symbole=CODE1&fields=  
/response/CodeValeur/Cours/text()+/response/CodeValeur/Variation/text()*

25 Selon le second mode de réalisation, la requête relative à l'exemple peut être formulée de la façon suivante :

```

<get_Data xmlns=urn:anyRR:transport:2005 ...>
  <Query> <!-- liste des paramètres dans la requête -->
    <ParameterGroup BinaryBooleanOperator="AND">
      <Parameter name="symbole" operator="EQ" value="CODE1+CODE2"/>
30 <Parameter name="langue" operator="EQ" value="fr"/>
    </ParameterGroup>
  </Query>
  <Fields> <!-- liste des champs demandés dans la réponse -->
    <Field name="/response/NomValeur/text()"/>
35 <Field name="/response/CodeValeur/text()"/>
    <Field name="/response/Cours/text()"/>
    <Field name="/response/Lien/text()"/>
  </Fields>
</get_Data>
40

```

Toutefois, pour que les informations 38 définissant les structures de requête 40 et de réponse 42 soient accessibles par un utilisateur qui n'est pas familier des langages informatiques, de préférence, le procédé comprend une étape 50 de présentation au

cours de laquelle le client génère, à l'aide de moyens de présentation 20, une fenêtre contenant par exemple une page HTML, formant une interface donnant accès ou présentant des informations fournies au client 12 par le serveur 14, plus particulièrement présentant les définitions des structures 40, 42.

5 Ces moyens de présentation 20 comprennent un programme d'interface non spécifique au serveur 14, téléchargé et installé au préalable sur le client. L'envoi au client 12 du programme d'interface peut être déclenché à la suite d'une commande de téléchargement volontaire du client 12.

10 L'envoi au client 12 du fichier de définition avec son type MIME peut être déclenché par l'activation du lien prévue à l'étape 32.

L'exécution du programme d'interface par le client 12 est déclenchée automatiquement par la réception du fichier de définition envoyé par le serveur 14 qui comporte un type MIME ( par exemple :« application/x-QRD-program ») associé à ce programme. Par exemple, si le programme informatique est destiné à exécuter les fichiers de format « .QRD », l'extension du fichier de définition est « .QRD » et le type MIME correspondant envoyé est alors « application/x-QRD-program ».Ce programme extrait les informations 38 contenues dans le fichier de définition pour créer, dans la page HTML de présentation affichée à l'écran du client 12, des éléments clairs décrivant les informations relatives aux structures de requête 40 et de réponse 42.

20 Les éléments descriptifs des structures 40, 42 apparaissant sur la page Internet sont par exemple des listes de valeurs ou un texte présentant les différents paramètres dans ce mode de réalisation.

25 De préférence, le programme d'interface crée également, dans la page HTML de présentation affichée à l'écran du client 12, des éléments d'assistance à la formulation de requêtes.

Ainsi, au moins un des éléments descriptifs de la page HTML de présentation peut comprendre une liste d'informations susceptibles de faire partie d'une réponse, les éléments d'assistance à la formulation d'une requête permettant de réaliser un choix parmi les informations de cette liste, par exemple à l'aide de boutons ou d'un champ de saisie.

30 Le programme d'interface est également utilisé pour décider de la destination des éléments contenus dans la réponse avec conversion éventuelle au codage utilisé chez le client.

35 L'étape 50 de présentation d'informations est suivie d'une étape 52 au cours de laquelle l'utilisateur client formule une requête à l'attention du serveur 14.

Cette étape 52 peut comprendre une sous-étape 54 de formulation de la requête à l'aide des éléments d'assistance à la formulation de requête proposés par le programme d'interface et de moyens classiques tels qu'un élément d'interface 22 du client 12 (clavier, souris, etc.).

5 L'étape 52 peut également comprendre une sous-étape 56 d'incorporation dans la requête de données propres au client 12, plus particulièrement de données contenues dans la base 24 du client 12.

La sous-étape 56 peut être exécutée notamment à l'aide de moyens comportant un logiciel utilitaire non spécifique au serveur 14 stocké sur l'unité centrale 20 du client 12.  
10 Après avoir formulé la requête, le client 12 émet cette requête vers le serveur 14, au cours d'une étape 58. Cette transmission peut se faire à la suite d'une action volontaire de l'utilisateur sur le client 12 (sous-étape 60 de l'étape 58) ou automatiquement (sous-étape 62 de l'étape 58).

Ainsi, au cours de la sous-étape 60, l'utilisateur client peut déclencher manuellement  
15 la transmission de la requête par exemple en cliquant sur un bouton de la fenêtre de présentation. Au cours de la sous-étape 60, la transmission de la requête est pilotée par des moyens classiques du client comprenant un élément 22 d'interface utilisateur (clavier, souris, etc.) et l'unité centrale 20 du client 12. La requête est transmise au serveur 14 par l'intermédiaire du réseau 16.

20 Au cours de la sous-étape 62, la requête est émise automatiquement par le client 12 vers le serveur 14, par exemple conformément à une séquence programmée par l'utilisateur dans le client 12.

Par exemple, au cours de la sous-étape 62, un formulaire est présenté à l'utilisateur client. Celui-ci indique, en remplissant des champs de saisies ou en cliquant sur des  
25 boutons numériques, à quelles échéances la requête doit être transmise automatiquement au serveur 14. Ce formulaire peut être généré par des moyens comprenant un logiciel utilitaire, par exemple le programme d'interface décrit ci-dessus, stocké dans l'unité centrale 20 du client 12.

30 Au cours de la sous-étape 62, la transmission de la requête est pilotée par des moyens classiques comprenant l'unité centrale 20 du client 12. La requête est transmise au serveur 14 par l'intermédiaire du réseau 16.

Après réception de la requête par le serveur 14, celui-ci la traite et transmet une réponse au client 12 au cours d'une étape 64. La réponse est conforme à la structure de réponse 42, incluse dans le fichier de définition présenté au préalable à l'utilisateur client.  
35 Dans le cas de la structure de réponse 42 donnée en exemple, la réponse selon le premier mode de réalisation peut être la suivante :

-19-

```

5 <response DateHeure="2005-02-01T16:41:00">
  <NomValeur>SOCIETE1</NomValeur>
  <CodeValeur>CODE1</CodeValeur>
  <Cours>23.25</Cours>
  <Conseil>3</Conseil>
</response>

```

Selon le second mode de réalisation, la réponse à la requête donnée en exemple est :

```

10 <?xml version="1.0" encoding="UTF-8"?>
  <get_Data_Result ...>
  <response>
    <NomValeur>SOCIETE1</NomValeur>
    <CodeValeur>CODE1</CodeValeur>
    <Cours>23.25</Cours>
15  <Lien>http:// adresse_serveur/accesSOAP/symbole=CODE1</Lien>
  </response>
  </get_Data_Result ...>

```

La réponse peut également être constituée par une arborescence d'éléments permettant de fournir plusieurs données distinctes, si bien qu'un seul envoi par le serveur 14 peut contenir toutes les données requises par le client 12. Le cas échéant, certaines données figurant dans la réponse du serveur 14 et provenant de la base 28 de ce serveur 14 sont affectées, avec conversion éventuelle au codage utilisé chez le client, au cours d'une étape 66, à la base de données 24 du client 12. Cette affectation peut être réalisée par un logiciel utilitaire stocké sur l'unité centrale 20 du client 12.

Le système et le procédé décrits ci-dessus permettent donc à un utilisateur client de générer une requête vers un serveur 14 et d'exploiter au mieux sa réponse quelle que soit la structure de données en vigueur sur ce serveur 14.

Le système d'échange de données selon l'invention ne se limite pas au mode de réalisation décrit ci-dessus.

En particulier, comme illustré en pointillés sur la figure 1, le système peut comprendre deux serveurs, à savoir le serveur 14 en trait plein (premier serveur) et le serveur 14' en pointillés (second serveur) plutôt qu'un seul et même serveur comme décrit précédemment. Dans cette configuration, le premier serveur 14 fournit au client des informations relatives aux structures de requête et de réponse conformes à la structure de données en vigueur sur le second serveur 14'.

## REVENDICATIONS

1. Procédé d'échange de données entre au moins un premier serveur (14) et un client (12), le procédé étant du type comprenant une étape (52) de formulation d'une requête à l'attention d'un second serveur (14, 14') qui, le cas échéant, peut être le premier serveur, fournissant des données structurées, notamment de type XML, **caractérisé en ce qu'il** comprend une étape préalable (36) de fourniture par le premier serveur (14) au client (12) d'informations relatives aux structures de requête (40) et de réponse (42) conformes à la structure de données en vigueur sur le second serveur (14, 14').

2. Procédé d'échange de données structurées selon la revendication 1, caractérisé en ce que les informations (38) relatives à la structure de requête (40), fournies au client (12), comprennent au moins une définition choisie parmi une définition (44a, 44b) d'un protocole de transfert de la requête, une définition d'une adresse de serveur, une définition d'une commande, et une définition (46) d'au moins un paramètre identifié par un nom et un type.

3. Procédé d'échange de données structurées selon la revendication 2, caractérisé en ce que les informations (38) relatives à la structure de requête (40), fournies au client (12), peuvent comprendre :

- un groupe d'au moins deux paramètres, mis en relation à l'aide d'opérateurs logiques ;
- pour chaque paramètre, un commentaire descriptif, une liste de valeurs attendues, une valeur par défaut et, si ce paramètre est de type numérique, une unité, une valeur minimale et une valeur maximale;
- une liste d'éléments et d'attributs à inclure dans la réponse.

4. Procédé d'échange de données structurées selon les revendications 1 à 3, caractérisé en ce que les informations (38) relatives à la structure de réponse (42), fournies au client (12), comprennent au moins une définition choisie parmi une définition d'un nom (48) de structure de données attendue en réponse et une définition d'au moins un élément de la réponse.

5. Procédé d'échange de données structurées selon la revendication 4, caractérisé en ce que les informations (38) relatives à la structure de réponse (42), fournies au client (12), peuvent comprendre, pour chaque élément, un nom, un type, une liste de valeurs possibles, un commentaire descriptif et, si l'élément est de type numérique, une unité et une cardinalité, associé éventuellement à un ou plusieurs attributs, avec, pour

chaque attribut, un nom, un type, une liste de valeurs possibles, un commentaire descriptif et, si l'attribut est de type numérique, une unité et une cardinalité.

5 6. Procédé d'échange de données structurées selon l'une quelconque des revendications 1 à 5, caractérisé en ce que l'étape (36) de fourniture d'informations est complétée par une étape de présentation (50) de ces informations à l'aide d'un programme d'interface exécuté par le client (12).

7. Procédé d'échange de données structurées selon la revendication 6, caractérisé en ce que le programme d'interface génère une fenêtre, contenant par exemple une page HTML, présentant ces informations.

10 8. Procédé d'échange de données structurées selon la revendication 6 ou 7, caractérisé en ce que le programme d'interface présente, d'une part, des éléments descriptifs décrivant les informations (38) relatives aux structures de requête (40) et de réponse (42) et, d'autre part, des éléments d'assistance à la formulation d'une requête.

15 9. Procédé d'échange de données structurées selon la revendication 8, caractérisé en ce qu'au moins un des éléments descriptifs comprend une liste d'informations susceptibles de faire partie d'une réponse, les éléments d'assistance à la formulation d'une requête permettant de réaliser un choix parmi les informations de cette liste.

20 10. Procédé d'échange de données structurées selon l'une quelconque des revendications 1 à 9, caractérisé en ce que l'étape (52) de formulation de requête comprend l'incorporation (56) dans la requête de données propres au client.

11. Procédé d'échange de données structurées selon l'une quelconque des revendications 1 à 10, caractérisé en ce qu'il comprend une étape (60) d'émission de la requête vers le second serveur (14, 14') à la suite d'une action volontaire d'un utilisateur sur le client (12).

25 12. Procédé d'échange de données structurées selon l'une quelconque des revendications 1 à 10, caractérisé en ce qu'il comprend une étape (62) d'émission automatique de la requête par le client (12) vers le second serveur (14, 14'), par exemple conformément à une séquence programmée par un utilisateur dans le client (12).

30 13. Procédé d'échange de données structurées selon l'une quelconque des revendications 1 à 12, caractérisé en ce que l'étape (36) de fourniture d'informations est déclenchée par une étape (32) d'activation volontaire d'un lien dans une page Internet proposée par le premier serveur (14).

35 14. Procédé d'échange de données structurées selon l'une quelconque des revendications 1 à 13, caractérisé en ce qu'il comprend une étape (64) de transmission

au client (12) par le second serveur (14, 14') d'une réponse à la requête émise par ledit client (12), la réponse étant conforme à la structure de réponse (42).

5 15. Procédé de fourniture de données par un serveur (14), **caractérisé en ce qu'il** comprend une étape de fourniture à un client (12) d'informations (38) relatives aux structures de requête (40) et de réponse (42) conformes à la structure de données en vigueur sur un serveur (14, 14') fournisseur de données structurées.

10 16. Serveur **caractérisé en ce qu'il** comprend des moyens (26) de fourniture à un client (12) d'informations (38) relatives aux structures de requête (40) et de réponse (42) conformes à la structure de données en vigueur sur un serveur (14, 14') fournisseur de données structurées.

17. Serveur selon la revendication 16, caractérisé en ce qu'il forme le serveur (14) fournisseur de données structurées.

15 18. Système comprenant au moins un premier serveur (14) et un client (12) reliés entre eux au moyen d'un réseau (16), le client (12) comportant des moyens (22) de formulation de requête à l'attention d'un second serveur (14, 14'), qui, le cas échéant, peut être le premier serveur, **caractérisé en ce que** le premier serveur (14) est selon la revendication 15.

20 19. Client d'un système selon la revendication 18, **caractérisé en ce qu'il** comprend des moyens de présentation (20) des informations (38) fournies par le premier serveur (14).

20. Client selon la revendication 19, caractérisé en ce qu'il comprend des moyens d'incorporation (20) de données qui lui sont propres dans la requête à l'attention du second serveur (14, 14').

25 21. Client selon la revendication 19 ou 20, caractérisé en ce qu'il comprend des moyens d'émission (20, 22) de la requête vers le second serveur (14, 14'), automatiquement ou à la suite d'une action volontaire d'un utilisateur.

30 22. Client selon l'une quelconque des revendications 19 à 21, caractérisé en ce qu'il comprend des moyens d'activation (20, 22) d'un lien dans une page Internet proposée par le premier serveur (14), déclenchant la fourniture des informations (38) par ledit serveur (14) au client (12).

23. Dispositif de fourniture de données, **caractérisé en ce qu'il** comprend des moyens (26) de fourniture à un client (12) d'informations (38) relatives aux structures de requête (40) et de réponse (42) conformes à la structure de données en vigueur sur un serveur (14, 14') fournisseur de données structurées.

35 24. Programme pour un ordinateur formant serveur (14) selon la revendication 16, **caractérisé en ce qu'il** fournit automatiquement au client (12), suite à l'activation d'un

lien par ledit client (12), des informations (38) relatives aux structures de requête (40) et de réponse (42) conformes à la structure de données d'un second serveur (14, 14').

5 25. Programme pour un ordinateur formant un client (12) d'un système selon la revendication 18, **caractérisé en ce qu'il** génère une interface de présentation d'informations fournies au client (12) par le premier serveur (14).

26. Programme d'ordinateur selon la revendication 25, caractérisé en ce que l'interface générée comprend au moins une fenêtre , contenant par exemple une page HTML, donnant accès ou présentant les informations fournies au client (12) par le premier serveur (14).

10 27. Programme d'ordinateur selon la revendication 25 ou 26, caractérisé en ce que l'interface générée présente, d'une part, des éléments descriptifs décrivant les informations relatives aux structures de requête (40) et de réponse (42) et, d'autre part, des éléments d'assistance à la formulation d'une requête.

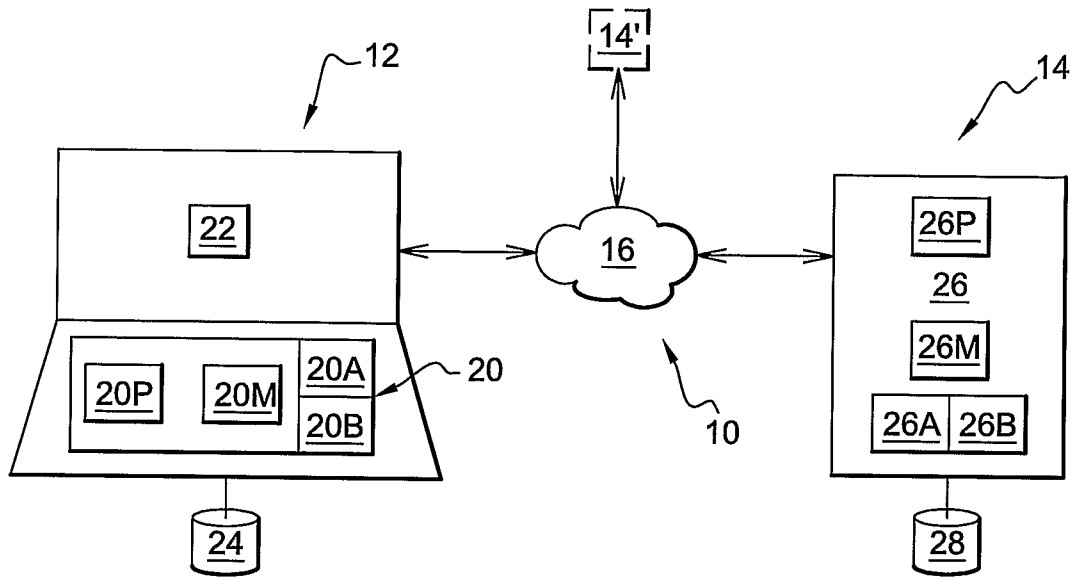


Fig. 1

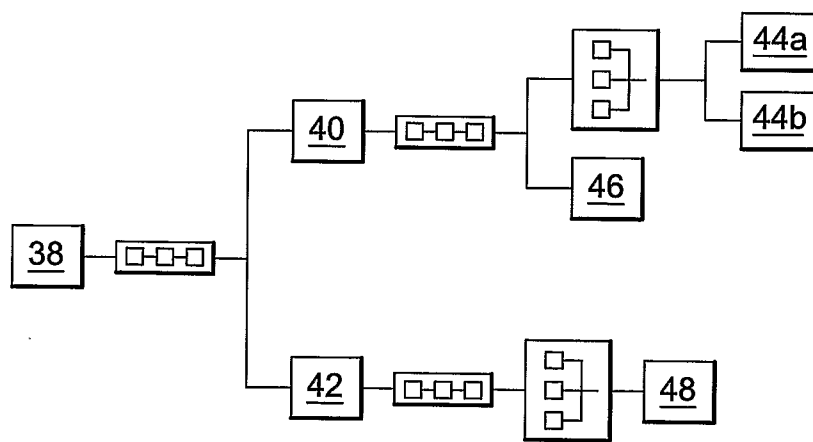
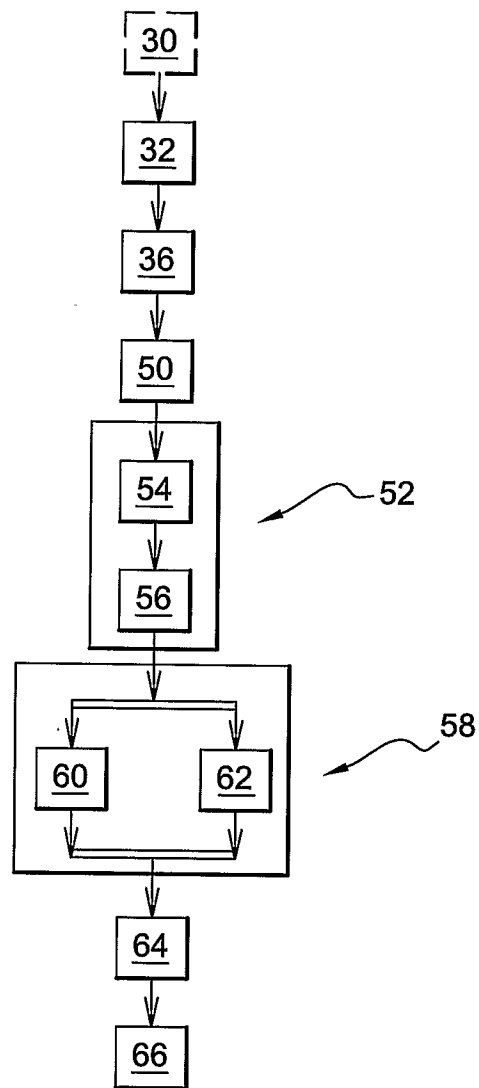


Fig. 3



**Fig. 2**

INTERNATIONAL SEARCH REPORT

International application No  
PCT/FR2006/001530

**A. CLASSIFICATION OF SUBJECT MATTER**  
INV. G06F17/30

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, INSPEC, COMPENDEX, IBM-TDB

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	<p>AGARWAL S ET AL: "Surfing the Service Web" SECOND INTERNATIONAL SEMANTIC WEB CONFERENCE, [Online] vol. 2870, October 2003 (2003-10), pages 211-226, XP002370002 Retrieved from the Internet: URL:http://citeseer.ist.psu.edu/agarwal03surfing.html&gt; [retrieved on 2006-02-27] abstract section 1. "Introduction" section 2. "Use Case" section 3. "Overview of the Complete Process of OntoMat-Service" section 7. "Web Services Invocation"</p> <p style="text-align: center;">----- -/--</p>	1-27

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	"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
"E" earlier document but published on or after the international filing date	"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
"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)	"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.
"O" document referring to an oral disclosure, use, exhibition or other means	"&" document member of the same patent family
"P" document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search <b>22 November 2006</b>	Date of mailing of the international search report <b>30/11/2006</b>
--	---

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 <b>LECHENNE-STILLER, L</b>
---	--

## INTERNATIONAL SEARCH REPORT

International application No  
PCT/FR2006/001530

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	<p>MRISSA M ET AL: "A Mediation Framework for Web Services in a Distributed Healthcare Information System"  MEDICAL INFORMATION SYSTEMS: THE DIGITAL HOSPITAL, 2004. IDEAS '04-DH. PROCEEDINGS. IDEAS WORKSHOP ON BEIJING, CHINA 01-03 SEPT. 2004, PISCATAWAY, NJ, USA,IEEE, 1 September 2004 (2004-09-01), pages 15-22, XP010779126  ISBN: 0-7695-2289-0  section 3.3 "Invocation"</p>	1
X	<p>JANECEK J: "Efficient SOAP processing in embedded systems"  ENGINEERING OF COMPUTER-BASED SYSTEMS, 2004. PROCEEDINGS. 11TH IEEE INTERNATIONAL CONFERENCE AND WORKSHOP ON THE BRNO, CZECH REPUBLIC 24-27 MAY 2004, PISCATAWAY, NJ, USA,IEEE, 24 May 2004 (2004-05-24), pages 128-134, XP010711447  ISBN: 0-7695-2125-8  section 2. "Classical web service"</p>	1
A	<p>WO 2005/052811 A (INTERNATIONAL BUSINESS MACHINES CORPORATION; IBM UNITED KINGDOM LIMITE) 9 June 2005 (2005-06-09)  paragraph [0044] - paragraph [0049];  figures 1,2  paragraph [0054] - paragraph [0057]  paragraph [0102] - paragraph [0104];  figure 4  paragraph [0121] - paragraph [0132]  paragraph [0167]</p>	1-27

# INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/FR2006/001530

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2005052811 A	09-06-2005	CN 1622086 A US 2005120016 A1	01-06-2005 02-06-2005

---

# RAPPORT DE RECHERCHE INTERNATIONALE

Demande internationale n°

PCT/FR2006/001530

<b>A. CLASSEMENT DE L'OBJET DE LA DEMANDE</b> INV. G06F17/30				
Selon la classification internationale des brevets (CIB) ou à la fois selon la classification nationale et la CIB				
<b>B. DOMAINES SUR LESQUELS LA RECHERCHE A PORTE</b>				
Documentation minimale consultée (système de classification suivi des symboles de classement) G06F				
Documentation consultée autre que la documentation minimale dans la mesure où ces documents relèvent des domaines sur lesquels a porté la recherche				
Base de données électronique consultée au cours de la recherche internationale (nom de la base de données, et si cela est réalisable, termes de recherche utilisés) EPO-Internal, WPI Data, INSPEC, COMPENDEX, IBM-TDB				
<b>C. DOCUMENTS CONSIDERES COMME PERTINENTS</b>				
Catégorie*	Identification des documents cités, avec, le cas échéant, l'indication des passages pertinents	no. des revendications visées		
X	AGARWAL S ET AL: "Surfing the Service Web" SECOND INTERNATIONAL SEMANTIC WEB CONFERENCE, [Online] vol. 2870, octobre 2003 (2003-10), pages 211-226, XP002370002 Extrait de l'Internet: URL: <a href="http://citeseer.ist.psu.edu/agarwal03surfing.html">http://citeseer.ist.psu.edu/agarwal03surfing.html</a> [extrait le 2006-02-27] abrégé section 1. "Introduction" section 2. "Use Case" section 3. "Overview of the Complete Process of OntoMat-Service" section 7. "Web Services Invocation" ----- -/--	1-27		
<table border="0" style="width: 100%;"> <tr> <td style="width: 50%; vertical-align: top;"> <input checked="" type="checkbox"/> Voir la suite du cadre C pour la fin de la liste des documents                 </td> <td style="width: 50%; vertical-align: top;"> <input checked="" type="checkbox"/> Les documents de familles de brevets sont indiqués en annexe                 </td> </tr> </table>			<input checked="" type="checkbox"/> Voir la suite du cadre C pour la fin de la liste des documents	<input checked="" type="checkbox"/> Les documents de familles de brevets sont indiqués en annexe
<input checked="" type="checkbox"/> Voir la suite du cadre C pour la fin de la liste des documents	<input checked="" type="checkbox"/> Les documents de familles de brevets sont indiqués en annexe			
* Catégories spéciales de documents cités:				
"A" document définissant l'état général de la technique, non considéré comme particulièrement pertinent "E" document antérieur, mais publié à la date de dépôt international ou après cette date "L" document pouvant jeter un doute sur une revendication de priorité ou cité pour déterminer la date de publication d'une autre citation ou pour une raison spéciale (telle qu'indiquée) "O" document se référant à une divulgation orale, à un usage, à une exposition ou tous autres moyens "P" document publié avant la date de dépôt international, mais postérieurement à la date de priorité revendiquée	"T" document ultérieur publié après la date de dépôt international ou la date de priorité et n'appartenant pas à l'état de la technique pertinent, mais cité pour comprendre le principe ou la théorie constituant la base de l'invention "X" document particulièrement pertinent; l'invention revendiquée ne peut être considérée comme nouvelle ou comme impliquant une activité inventive par rapport au document considéré isolément "Y" document particulièrement pertinent; l'invention revendiquée ne peut être considérée comme impliquant une activité inventive lorsque le document est associé à un ou plusieurs autres documents de même nature, cette combinaison étant évidente pour une personne du métier "&" document qui fait partie de la même famille de brevets			
Date à laquelle la recherche internationale a été effectivement achevée 22 novembre 2006		Date d'expédition du présent rapport de recherche internationale 30/11/2006		
Nom et adresse postale de l'administration chargée de la recherche internationale Office Européen des Brevets, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016		Fonctionnaire autorisé LECHENNE-STILLER, L		

RAPPORT DE RECHERCHE INTERNATIONALE

Demande internationale n°  
PCT/FR2006/001530

C(suite). DOCUMENTS CONSIDERES COMME PERTINENTS		
Catégorie*	Identification des documents cités, avec, le cas échéant, l'indication des passages pertinents	no. des revendications visées
X	<p>MRISSA M ET AL: "A Mediation Framework for Web Services in a Distributed Healthcare Information System"                      MEDICAL INFORMATION SYSTEMS: THE DIGITAL HOSPITAL, 2004. IDEAS '04-DH. PROCEEDINGS. IDEAS WORKSHOP ON BEIJING, CHINA 01-03 SEPT. 2004, PISCATAWAY, NJ, USA, IEEE, 1 septembre 2004 (2004-09-01), pages 15-22, XP010779126                      ISBN: 0-7695-2289-0                      section 3.3 "Invocation"</p>	1
X	<p>JANECEK J: "Efficient SOAP processing in embedded systems"                      ENGINEERING OF COMPUTER-BASED SYSTEMS, 2004. PROCEEDINGS. 11TH IEEE INTERNATIONAL CONFERENCE AND WORKSHOP ON THE BRNO, CZECH REPUBLIC 24-27 MAY 2004, PISCATAWAY, NJ, USA, IEEE, 24 mai 2004 (2004-05-24), pages 128-134, XP010711447                      ISBN: 0-7695-2125-8                      section 2. "Classical web service"</p>	1
A	<p>WO 2005/052811 A (INTERNATIONAL BUSINESS MACHINES CORPORATION; IBM UNITED KINGDOM LIMITE) 9 juin 2005 (2005-06-09)                      alinéa [0044] - alinéa [0049]; figures 1,2                      alinéa [0054] - alinéa [0057]                      alinéa [0102] - alinéa [0104]; figure 4                      alinéa [0121] - alinéa [0132]                      alinéa [0167]</p>	1-27

# RAPPORT DE RECHERCHE INTERNATIONALE

Renseignements relatifs aux membres de familles de brevets

Demande internationale n°

PCT/FR2006/001530

Document brevet cité au rapport de recherche	Date de publication	Membre(s) de la famille de brevet(s)	Date de publication
WO 2005052811 A	09-06-2005	CN 1622086 A	01-06-2005
		US 2005120016 A1	02-06-2005
-----			