

**(19) AUSTRALIAN PATENT OFFICE**

- (54) Title  
Method, software application and system for exchanging benchmarks
- (51)<sup>6</sup> International Patent Classification(s)  
**G06F** 11/34 (2006.01) 15/00  
**G06F** 11/36 (2006.01) 20060101ALI2006031  
**G06F** 12/00 (2006.01) 08BMEP **G06F**  
**G06F** 15/00 (2006.01) 17/22  
**G06F** 17/22 (2006.01) 20060101ALI2005100  
**G06F** 17/27 (2006.01) 8BMEP **G06F**  
**G06F** 17/30 (2006.01) 17/27  
**G06Q** 10/00 (2006.01) 20060101ALI2005100  
**G06Q** 30/00 (2006.01) 8BMEP **G06F**  
**G06F** 11/34 17/30  
20060101AFI2006031 20060101ALI2006031  
08BMEP **G06F** 08BMEP **G06Q**  
11/36 10/00  
20060101ALI2006031 20060101ALI2005100  
08BMEP **G06F** 8BMEP **G06Q**  
12/00 30/00  
20060101ALI2006031 20060101ALI2005100  
08BMEP **G06F** 8BMEP  
PCT/EP03/01436
- (21) Application No: 2003206891 (22) Application Date: 2003 .02 .13
- (87) WIPO No: W003/069522
- (30) Priority Data
- |             |              |              |
|-------------|--------------|--------------|
| (31) Number | (32) Date    | (33) Country |
| 60/355,840  | 2002 .02 .13 | US           |
- (43) Publication Date : 2003 .09 .04  
(43) Publication Journal Date : 2003 .10 .02
- (71) Applicant(s)  
**SAP AG**
- (72) Inventor(s)  
Fleckenstein, Thomas; Wefers, Marcus
- (74) Agent/Attorney  
Griffith Hack, Level 29, Northpoint 100 Miller Street, North Sydney, NSW, 2060

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

(19) World Intellectual Property Organization  
International Bureau



(43) International Publication Date  
21 August 2003 (21.08.2003)

PCT

(10) International Publication Number  
**WO 03/069522 A2**

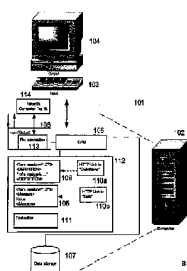
- (51) International Patent Classification<sup>7</sup>: **G06F 17/60**
- (21) International Application Number: PCT/EP03/01436
- (22) International Filing Date: 13 February 2003 (13.02.2003)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:  
60/355,840 13 February 2002 (13.02.2002) US
- (71) Applicant (for all designated States except US): **SAP AKTIENGESELLSCHAFT** [DE/DE]; Neurottstr. 16, 69190 Walldorf (DE).
- (72) Inventors; and
- (75) Inventors/Applicants (for US only): **WEFERS, Marcus** [DE/DE]; Im Lindenried 13, 69118 Heidelberg (DE).
- (74) Common Representative: **SAP AKTIENGESELLSCHAFT**; Intellectual Property Department, Dr. Wolfgang Popp, Neurottstr. 16, 69190 Walldorf (DE).
- (81) Designated States (national): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SL, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.
- (84) Designated States (regional): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, SE, SI,

[Continued on next page]

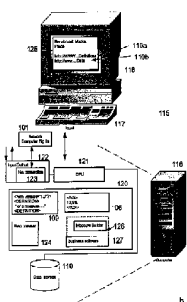
(54) Title: METHOD, SOFTWARE APPLICATION AND SYSTEM FOR EXCHANGING BENCHMARKS



WO 03/069522 A2



(57) Abstract: The invention relates to an electronic data structure comprising a benchmark definition, said electronic data structure being a file having a format readable by a SGML-based language. The invention further relates to a method for processing and storing data objects by means of one or more processes running in a computer system having one or more of the electronic data structures comprising a step of receiving one or more of said electronic data structures by electronic means.





SK, TR), OAPI patent (BE, BJ, CE, CG, CI, CM, GA, GN, CQ, GW, ML, MR, NE, SN, TD, TG).

*For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.*

**Published:**

— *without international search report and to be republished upon receipt of that report*

Method, Software Application and System for Exchanging  
Benchmarks

### Background of the Invention

#### Field of the Invention.

5 The technical field of this invention is in the area of electronic data processing. More particularly, the invention relates to methods, computer program products and systems for automated exchange of benchmarks and benchmark data.

#### 10 Description of the Related Art

A benchmark, in general, is a point of reference by which something can be measured. In surveying, a "benchmark" (two words) is a post or other permanent mark established at a known elevation that is used as the  
15 basis for measuring the elevation of other topographical points.

In economic environment, particularly in computer and Internet technology, "benchmark" may have any of these meanings:

20 A set of conditions against which a product or system is measured. PC magazine laboratories frequently test and compare several new computers or computer devices against the same set of application programs, user interactions, and contextual situations. The total  
25 context against which all products are measured and compared is referred to as the benchmark.  
A program that is specially designed to provide measurements for a particular operating system or application.

- 1 -

CONFIRMATION COPY

A known product with which users are familiar or accustomed to that other newer products can be compared to.

5 A set of performance criteria, which a product is expected to meet.

Within the concept of this invention, benchmarks are standards or averages by which similar items can be compared, and benchmark definitions comprise rules for the calculation and interpretation of benchmark data.

10

Benchmark data has become a critical input factor for performance measurement and strategy management of enterprises. This information is not only used for measurement purposes but also essential to enable a meaningful planning processes. Companies need quality information about their competitors, industry, products, markets etc. Benchmarks can be useful to satisfy such needs, and some companies offer benchmark programs for downloading or a benchmark testing service on their own web site.

20

Benchmark definitions and data exist in the economy on a horizontal level, e.g. for areas like human resources, and on a vertical level, e.g. focused on industry, products, services, performances. Benchmarks are hereinafter alternatively referred to as "measures".

25

However, if a company intends to perform benchmark analyses for its products or services or its performance with respect to its competitors, the company has to get the benchmark definitions or data on its own or has to consult a benchmark provider. This is a time consuming task, even if the company uses an ERP software (enterprise resource planning), especially if

35

the company itself performs the task. Further, a lot of compatibility problems may arise if the data shall be incorporated into the company's ERP software, because each benchmark provider provides its own data format, which does not always fit to the format required by the company's ERP software. A further problem of getting data is that they are not publicly available.

It would be advantageous if there was provided a data structure, method, software application and/or data processing system providing a more efficient solution of the problems described above, and particularly a data structure and software application for an easy and comfortable exchange or incorporation of benchmarks and benchmark data into a users business software.

#### Summary of the Invention

In accordance with a first aspect of the present invention there is provided a method for processing electronic data structures, the electronic data structures each comprising a benchmark definition and having a file format that is readable by a SGML-based language, the method comprising the steps of: receiving one or more of said electronic data structures by electronic means; and mapping a benchmark definition contained in said electronic data structure to equivalent benchmark definition contained in software for supporting business processes by using a table, in which a first ID of the benchmark definition in the electronic data structure is assigned to a second ID of the benchmark definition in the software for the support of business processes.

In embodiments, by using the electronic data structure in data processing, benchmark data may be easily exchanged, e.g. between benchmark users and providers. Advantageously, embodiments of the present invention may also solve the technical problem of

5 establishing an easy path for exchanging electronic data on benchmarks between two computer systems.

In accordance with a second aspect of the invention, there is provided a computer system for processing electronic data structures, the

10 electronic data structures comprising a benchmark definition and having a file format that is readable by a SGML-based language, the system comprising: memory means having program instructions; input means for entering data; storage means for storing data; a processor responsive to the program instructions for receiving one or

15 more of said electronic data structures by electronic means; and means for mapping the benchmark definition contained in said electronic data structure to an equivalent benchmark definition contained in software for support of business processes by using a table, in which a first ID of the benchmark definition in the electronic

20 data structure is assigned to a second ID of the benchmark definition in the software for the support of business processes.

In accordance with a third aspect of the invention, there is provided a computer program product for processing electronic data structures,

25 the electronic data structures comprising a benchmark definition and having a file format that is readable by SGML-based language, the program comprising at least one instruction which, when implemented on a computer readable medium of a computing system, causes the computing system to implement the method steps

30 according to the first aspect.

- 4a -

In accordance with a fourth aspect of the invention, there is provided a computer readable medium providing a computer program product in accordance with the third aspect.

- 5 In accordance with a fifth aspect of the present invention, there is provided a computer data signal embodied in a computer readable medium comprising code for processing electronic data structures, the electronic data structures each comprising a benchmark definition and having a file format that is readable by a SGML-based language,  
10 said code comprising instructions for implementing the method according to the first aspect.

Additional objects and advantages of the invention will be set forth in part in the description, or may be learned by practice of the  
15 invention. The objects and advantages of the invention will be realized and attained by means of the elements and combinations particularly pointed out in the appended claims. Embodiments of the invention are disclosed in the detailed description section and in the dependent claims.

20 It is understood that both the foregoing general description and the following detailed description are exemplary and explanatory only and are not restrictive of the invention, as claimed.



### Brief Description of the Drawings

The accompanying drawings, which are incorporated in and constitute a part of this specification, illustrate embodiments of the invention and, together with the description, explain the principles of the invention.

5 In the drawings,

Fig. 1 is a schematic block diagram of the implementation of the inventive electronic data structure within a computer system.

10

Fig. 2 is a schematic block diagram of a scenario for selecting a benchmark provider.

15 Fig. 3 is a schematic block diagram of a scenario for selecting a benchmark package.

Fig. 4 is a schematic block diagram of a scenario for benchmark retrieval.

20

Fig. 5 is a schematic block diagram of a scenario for customer participation in benchmark study.

Fig. 6 is a schematic block diagram of a scenario for updating benchmark definitions.

25

Fig. 7 is an exemplary relation among different elements of a structure within an inventive electronic data structure.

30

### Detailed Description

Computer and program are closely related. As used hereinafter, phrases, such as "the computer provides" and "the program provides or performs specific actions", are convenient abbreviation to express actions by a computer that is controlled by a program or to express that the program or program module is designed to enable the computer to perform the specific action.

10

It should be understood that the term "presentment" as used herein does not include the specialized definition normally associated with commercial paper, i.e. the production on a negotiable instrument to a drawee.

15

Rather, the term refers to providing via electronic means an "electronic data structure". This electronic presentment may take place through the use of an internet website or e-mail or SMS, a bank ATM machine or through the use of a stand alone kiosk.

20

Reference will now be made in detail to the principles of the invention by explaining the invention on the basis of a data processing process, examples of which are illustrated in the accompanying drawings. Examples, mentioned therein, are intended to explain the invention and not to limit the invention in any kind.

25

Within the concept of this invention, the terms used shall have their usual meaning in the context of the field of data processing unless defined otherwise in the following section:

30

A computer system can be a stand alone computer such as a PC or a laptop or a series of computers connected as

a network, e.g. a network within a company, or a series of computers connected via the internet, including any usual peripheral devices, respectively.

- 5 A data structure is a specialized format for organizing and storing data. General data structure types include the array, the file, the record, the table, the tree, and so on. Any data structure is designed to organize data to suit a specific purpose so that it can be  
10 accessed and worked with in appropriate ways. In computer programming, a data structure may be selected or designed to store data for the purpose of working on it with various algorithms. Within the concept of this invention, an electronic data structure is a data  
15 structure stored on a volatile or nonvolatile memory or embedded in a carrier signal and accessible by a computer system.

- The term business software shall mean software or  
20 software applications or programs or program modules, which support business processes in the economy, particularly in enterprises.  
ID is the abbreviation for identifier.

- 25 SGML (Standard Generalized Markup Language) is a standard for how to specify a document markup language or tag set. Such a specification is itself a document type definition (DTD). SGML is not in itself a document language, but a description of how to specify one. It  
30 is metadata.  
SGML is based on the idea that documents have structural and other semantic elements that can be described without reference to how such elements should be displayed. The actual display of such a document may  
35 vary, depending on the output medium and style

preferences. Some advantages of documents based on SGML are:

They can be created by thinking in terms of document structure rather than appearance characteristics (which  
5 may change over time).

They will be more portable because an SGML compiler can interpret any document by reference to its document type definition (DTD).

Documents originally intended for the print medium can  
10 easily be re-adapted for other media, such as the computer display screen.

The language that this Web browser uses, Hypertext Markup Language (HTML), example of an SGML-based language, XML is another example. There is a document  
15 type definition for HTML (and reading the HTML specification is effectively reading an expanded version of the document type definition).

XML (Extensible Markup Language) is a flexible way to create common information formats and share both the  
20 format and the data on the World Wide Web, intranets, and elsewhere. XML can be used by any individual or group of individuals or companies that wants to share information in a consistent way.

XML, a formal recommendation from the World Wide Web  
25 Consortium, is similar to the language of today's Web pages, the Hypertext Markup Language. Both XML and HTML contain markup symbols to describe the contents of a page or file. HTML, however, describes the content of a Web page (mainly text and graphic images) only in terms  
30 of how it is to be displayed and interacted with. This means that an XML file can be processed purely as data by a program or it can be stored with similar data on another computer or, like an HTML file, that it can be displayed.

XML is "extensible" because the markup symbols are unlimited and self-defining. XML is actually a simpler and easier-to-use subset of the Standard Generalized Markup Language, the standard for how to create a document structure. HTML and XML may be used together in many Web applications. XML markup, for example, may appear within an HTML page.

A first embodiment of the electronic data structure is characterized in that said file is an XML file. An alternative embodiment of the electronic data structure is characterized in that said file is an HTML file. A further embodiment is characterized in that the file further comprises benchmark data.

A first embodiment of the inventive method as described in the summary section is characterized in that the method further comprises a step of the electronic means comprise the internet and wherein a link to said one or more of the electronic data structures is presented on an internet page, which link is used to receive one or more of the electronic data structures.

A second embodiment of the inventive method is characterized in that the method further comprises receiving one or more of said electronic data structures by email or SMS.

A third embodiment comprises receiving one or more of said electronic data structures by using the http or ftp protocol.

A further embodiment comprises incorporating one or more of the benchmark definition or benchmark data

contained in said one or more electronic data structures into a software for support of business processes.

- 5 A further embodiment is characterized by the incorporation of the benchmark definition is performed automatically, comprising the steps of:  
reading one or more of said electronic data structures,  
recognizing one or more key words for benchmark  
10 definition within one or more of said electronic data structures,  
writing information, identified by one or more of said key words, into a data base structure.
- 15 A further embodiment is characterized in that the incorporation of the benchmark data is performed automatically, comprising the steps of:  
reading one or more of said electronic data structures,  
recognizing one or more key words for benchmark data  
20 within one or more of said electronic data structures,  
writing information, identified by one or more of said key words, into a data base structure.

In a still further embodiment the invention comprises  
25 the information, identified by a key word, and the data base structure, into which said information is written, are assigned to each other by an algorithm.

Additionally, a still further embodiment comprises said  
30 algorithm comprises one or more tables or one or more rules.

An other embodiment is characterized by said one or more rules are contained in or more tables.

35

In a further embodiment of the inventive method, mapping a benchmark definition contained in said electronic data structure to an equivalent benchmark definition contained in said software for support of business processes by means of a table, in which a first ID of the benchmark definition in the electronic data structure is assigned to a second ID of the benchmark definition in the software for support of business processes.

10

A still further embodiment is a the inventive method for use in a software for support of business processes, particularly in an enterprise resource planning software.

15

Processors suitable for the execution of a computer program include, by way of example, both general and special purpose microprocessors, and any one or more processors of any kind of digital computer. Generally, a processor will receive instructions and data from a read-only memory or a random access memory or both. The essential elements of a computer are a processor for executing instructions and one or more memory devices for storing instructions and data. Generally, a computer will also include, or be operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices (storage means) for storing data, e.g., magnetic, magneto-optical disks, or optical disks. Information carriers suitable for embodying computer program instructions and data include all forms of non-volatile memory, including by way of example semiconductor memory devices, such as EPROM, EEPROM, and flash memory devices; magnetic disks such as internal hard disks and removable disks;

20

25

30

35

- 11 -

magneto-optical disks; and CD-ROM and DVD-ROM disks.  
The processor and the memory can be supplemented by, or  
incorporated in, ASICs (application-specific integrated  
circuits).

5

To provide for interaction with a user, the invention  
can be implemented on a computer system having a  
display device such as a CRT (cathode ray tube) or LCD  
(liquid crystal display) monitor for displaying  
10 information to the user and a keyboard and a pointing  
device such as a mouse or a trackball by which the user  
can provide input to the computer. Other kinds of  
devices can be used to provide for interaction with a  
user as well; for example, feedback provided to the  
15 user can be any form of sensory feedback, such as  
visual feedback, auditory feedback, or haptic feedback;  
and input from the user can be received in any form,  
including acoustic, speech, or haptic input.

20 The invention and some of its possible embodiments is  
now described in more detail by way of reference to the  
drawings.

Figures 1a and 1b depict one example of an  
25 implementation of an embodiment of the invention: a  
computer system 101 connectable to a computer system  
115, each with program modules for performing the  
inventive method and its implementations. Figure 1 a  
shows a computer system 101 comprising a computer 102  
30 having a CPU 105, a working storage 112 (memory), in  
which software applications are stored for being  
processed by CPU 105. Such a software application may  
be a text editor 111, for creation of the inventive  
data structures: a file 109 comprising definitions of  
35 benchmarks and a file 106 comprising benchmark data.

- 12 -



Computer system 101 further comprises input means 103 and output means 104 for interaction with a user, e.g. for starting programs and/or data input and/or output. Computer system 101 further comprises general

5 input/output means 108, including a net connection 113, for sending and/or receiving data, e.g. for a net connection with one or more further computer systems 114, or for files like the inventive files 106, 109 from other parties. A plurality of computer systems

10 like 101, particularly a computer system 115 as shown in fig. 1b, may be connected via the net connection 113 in the form of the network 114 in such a case, the network computers 114 can be used as further

15 input/output means, including the use as further storage locations. For storing data, computer system 101 comprises a nonvolatile storage means 107. Figure 1b shows the computer system 115 connectable to the computer system 101 of figure 1 a. Computer system 115

20 comprises a computer 116 having a CPU 121, a working storage 120 (memory), in which software applications are stored for being processed by CPU 121, general input/output means 122, including a net connection 123, for sending and/or receiving data and for a net

25 connection to other computer systems, particularly to computer system 101 of figure 1 a. Computer system 115 further comprises input means 117 and output means 118 for interaction with a user, e.g. for starting programs and/or data input and/or output, and a nonvolatile storage means 119.

30

Within the hardware example of figure 1a and 1b, the inventive data structures 106, 109 are installed on the computer system 101. Links 110a, 110b are incorporated in a web page 125, which is accessible via the internet

35 and a web browser 124 and thus enables a second party

to process to download (receive) the files 106, 109 by means of computer system 115 when it is connected to computer system 101. The links 110a, 110b and the files 106, 109 may alternatively be sent a third party  
5 by other electronic means, like email, SMS.

After receipt of the files 106, 109 a software application 126 (hereinafter referred to as "measure builder") reads the definition file 109 and recognizes  
10 one or more key words for benchmark definitions. The definition identifies information about how to calculate the benchmark. The measure builder then writes the information, identified by one or more of said key words, into a data base structure for access  
15 for a software 127, which calculates and/or further processes benchmark data according to the benchmark definition. A user may select which benchmark definitions he wants to have incorporated into his software 127. If the benchmark definitions are then  
20 used by software 127 to calculate benchmark data, the calculated benchmark data may be based on business data of the respective company, which uses the system and which are contained in or accessible by software 127. The measure builder 126 may be part of the software  
25 127. In order to correctly recognize the benchmark definitions in file 109, the measure builder may have a list of valid benchmark definitions.

30 Within the following paragraphs, examples of further implementations of this inventions, possible interactions of a user of a computer system including the data structures and/or methods of the claimed invention ("customer") are described, without  
35 limitation of the invention in any kind.

## Terms used:

Benchmark provider A company that offers services to customers such as

- benchmark data, and benchmark definitions (measures)
- participation in benchmark studies,
- consulting,
- advanced analysis of benchmark data

Measure builder Software application for the definition, description and analysis of measures (benchmark definitions). The application offers in further implementations the maintenance and analysis of benchmark data as well.

Measure Catalog Customer defined catalog of measures maintained by the measure builder. A ready to use measures Catalogs is referred to as "Business Content"

5

In order to satisfy the needs of parties, which use the systems, methods or computer programs according to the present invention, systems, methods or computer programs as claimed may have additional features as

10 described in the following sections:

Thus, the invention comprises a method or module for incorporating external benchmarks into strategy management and performance measurement processes or

15 applications of customers.

The invention further comprises a method or module for the investigation of industry standard measures. This process can be supported by one or more measure catalogs (so-called delivered Business Content) or  
5 measure listings of benchmark providers, which have experience in the respective industry or area.

The invention further comprises a method or module for the identifying a subset of those measures meaningful  
10 for the company.

The invention further comprises a method or module for identifying appropriate benchmark provider for respective area; by industry, country, product or  
15 process.

The invention further comprises a method or module for making benchmark data available for analysis and comparison of actual and plan data against best-in-  
20 class, industry average, competitors. The availability of benchmark data for the analysis can be implemented in the following two complementary ways:

1. Transferring external benchmark data from  
25 benchmark provider into a benchmark storage. This enables the customer to use them within analytical applications such as a balanced scorecard, management cockpit, value driver trees, business planning & simulation, reporting, financial  
30 analytics, customer relationship management analytics, SCM analytics or human capital management analytics.
2. Accessing data directly via web-based services.  
35 Some benchmark provider offer in-depth analytical

services based on their large benchmark data base.  
Alternatively, provider may offer their data plus  
analytical tools via CD delivery for viewing at  
the customer site.

5

The invention further comprises a method or module for  
submitting data to benchmark provider for participation  
in provider supported survey. This includes the  
administration and creation of the exchange document

10 based on pre-configurable queries for standard  
measures.

The invention further comprises a method or module or a  
further means for enabling the listing of benchmark  
15 providers, which supports certain measures. Filters for  
benchmark provider may be available to provide overview  
of measure coverage on a provider basis.

The list of providers may be accessible in a convenient  
location, ideally linked from the measure builder or  
20 directly from a web page.

The invention further comprises a method or module or a  
means for enabling a customer to access a central place  
(e.g. internet market place), which provides  
information about benchmark providers that work with  
25 certain companies, scope of offering, category of  
offering (country, industry, special area, validity),  
web page and license information.

The provider list can be sorted by certain criteria  
such as industry, country or validity by further.

30 Information about packages that are available for  
download into the a customer's business software  
(Customer BS) system may be contained in the provider  
web page. The user can review those package offerings  
directly on the web page of the provider and select a

package. Such provider web page can be easily accessible.

In order to provide a legal basis for the relation between a provider and a customer, a license agreement  
5 may be signed directly between the customer and the provider. This agreement can define e.g. package, number of users, fees and validity.

The invention further comprises a method or module or a further means that enables the identification of users.

10 If named users are required, an easy to use way is provided to the user to communicate the names to the provider, who in return will submit password information etc directly to the user.

15 Some benchmark providers may require their customer to supply data prior to access information. This can be a very labor-intensive process, which can be shortened significantly by using data collection capabilities and predefined content. The following features support  
20 these requirements/specifications:

means for enabling a user to select a survey directly from benchmark provider web page,

means for enabling a review of measures prior to

25 transfer to provider,

means for transferring data to provider in an easy and secure fashion, e.g. by https encryption,

means for providing a transfer receipt to be sent from the provider to the user.

30

For retrieving benchmark data from provider to a customer's business software system, the invention further comprises a method or module or a means for:  
logon handling of provider website;

35 marking measures or grouping in categories;

- enabling secure transfer;
- the import of data into customer's business software system, supported by validation check and having viewing capability prior to update;
- 5 viewing the history of benchmark transfers.

For application and embedding of external benchmarks, the invention further comprises a method or module or a means for:

- 10 reporting of benchmark data in measure builder;
- using external benchmarks for analysis with customer's business software, wherein drill-down reporting for detailed analysis may be enabled;
- For satisfying needs of providers, the invention
- 15 further comprises a method or module or a means for:
- enabling a customer to send data to provider, e.g. to participate in survey;
- handling of mass requests, such as regular updates of data,
- 20 means for informing users of updated benchmark data as well as of benchmark definition;
- accessing measure ID list, what can serve as the basis to identify measures between a customer's business software system and provider;
- 25 promoting offering on web pages;
- associating measures with standard measure, e.g. to promote offering;
- enabling of drill-down reporting.
- 30 The following section describes examples of possible processes of the exchange of benchmark definitions and/or benchmark data between users of the invention with reference to the figures.

The following steps can be used as building blocks of the benchmark exchange.

Process 1:	Selection of provider
Process 2:	Selection of benchmark package
Process 3:	Transfer of benchmark package and update into benchmark storage
Process 4:	Customer Participation in benchmark study

5

Explanation of components used in Flow description:

Component	Description
Measure builder	Measure builder at Customer site
BW	Underlying business data information system for a customer business software system
XML file containing benchmark data	Document used for seamless exchange of benchmark data between provider and customer and vice versa.
Benchmarking Website	Listing of benchmark partner companies including information about their service offering. This sites allows to browse the measure available by each individual provider. Customers can download latest definitions and descriptions of available measures used for benchmarking.
Benchmark provider Website	Web-enabled benchmarking application that supports to sign contracts, logon, review offering , select benchmark packages, generate XML files, store XML files, download XML files.



Customer BS system	Software system of a customer for supporting business processes
Customer computer	Computer system, which the customer uses to perform the described actions.
Provider computer	Computer system, which the benchmark provider uses to perform the described actions.

#### Selection of provider

- 5 The steps for this process are described in the following table with reference to Fig. 2.

Step	Short description	Description and comments
1a	Logon	User logon to measure builder.
1b	Access service marketplace	Alternatively, the user accesses the service marketplace.
2	View list of benchmark providers	The user accesses the service marketplace that includes an additional page for the benchmark topic. Here the user finds a list of benchmark partner companies. Also, a list exists that allows to view measure definitions grouped by each benchmark provider.
3	Access benchmark provider website	User can review benchmark provider service offering, such as benchmark data subscription, consulting services. This might include viewing of sample data or limited access to benchmark data base. Contract preparation, such as pricing models, up to sign-off of contract

4	Benchmark access information	Provider informs customer about access details, like user ID's and passwords
---	------------------------------	--

#### Selection of benchmark package

The steps for this process are described in the

5 following table with reference to Fig.3.

Step	Short description	Description and comments
1	Logon	Customer logon to provider website and identification through User-ID and password. This logon can be accomplished by using a partner URL directly, but also via a partner link on a third party's benchmarking web page. This benchmarking web page can be accessed directly, from the measure builder.
2	Benchmark package compilation / selection	Selection of predefined benchmark package that for example includes multiple measures and a given time interval  Alternatively, some provider might offer an individual and dynamic measure selection process through their website
3	Package creation	Benchmark provider creates the requested package (XML file) and stores this on the providers web-site (technically it will be stored on a file server)
4	Customer notification	Customer will be notified by email when package is ready

5	Optional: View content	Provider may offer to view the content of the benchmark package (XML-file) directly on their website.
---	------------------------	---

Transfer of benchmark data and update benchmark storage

Customer retrieves benchmark package directly from  
 5 provider web service. As already described in the  
 selection process, the Customer can access provider's  
 website and choose desired benchmarks from a html page  
 of menu items. After selection, the provider can  
 generate a valid XML document according to the  
 10 selections of the customer and store this file on a  
 provider fileserver or data base.

The customer will then access the file via a provider  
 web service and download the file locally or to  
 15 customer fileserver.

The invention further provides parsing and viewing  
 capabilities of the file as well as a mechanism to  
 update the data to a benchmark storage by mapping the  
 20 XML structure to a benchmark data storage.

The software used by customer and provider with respect  
 to this process are designed such that

1. XML structure is reconciled between customer and  
 25 external provider company,
2. benchmark and measure identifiers are reconciled  
 between customer and external provider company,
3. customer can activate relevant measures in measure  
 builder,
- 30 4. customer can selected benchmark package directly on  
 provider's web page,

5. provider's software has means to generate requested benchmark package according to XML structure.

- A benchmarking service contract may have been put in place between customer and benchmark provider, in order to put the benchmark exchange business on a contractual basis.

- In an alternative implementation, provider may transfer benchmark file to a file server hosted by an third party. The customer then retrieves data from this files server for further processing and update into a benchmark data storage.

- 15 The implementation, in which a customer retrieves benchmark package directly from provider web service is described in the following table with reference to Fig. 4.

Step	Short description	Description and comments
1	Optional step: Logon	Customer logon to provider website and identification through User-ID and password. This logon can be accomplished by using the partner URL directly, but also via the partner link on a third party's benchmarking web page. That benchmarking web page can be accessed directly, from the measure builder or the third party's website. This is an optional step as it's not necessary for customer to download XML file from provider web site as upload program can read provider's website directly.

2	Optional step: Selection of XML-File	Customer selects XML file that contains requested benchmark package. This may be handled by selecting a file from a list or direct access to the file via an URL that the provider communicated to the customer.  The provider may also provide viewing capabilities for the selected package.
3	Optional step: Download XML File	Customer transfers file to storage location on customer side such as local hard drive or customer network folder. Alternatively, the file could also be sent via a CD or email, depending on the providers offerings. FTP could be used as well.
4	Logon	User logon to measure builder.
5	Optional: read benchmark XML file from provider website	Upload program read benchmark XML file from provider's website, thus user doesn't need to download file. But he must tell the upload program the URL of the benchmark XML file as well as user name and password(if required).
6	Preparation of benchmark Update	From the measure builder, the upload of the file will be initiated by selecting the file from either the local data storage or URL on provider website. The file may be validated, parsed and mapped to the benchmark data storage structure. benchmark data can be reviewed in order to verify accuracy and relevance prior to update of the benchmark data storage

7	Update	benchmark data will be updated into the benchmark storage.
---	--------	--

#### Customer participation in benchmark study

- 5 The steps for this process are described in the following table with reference to Fig. 5.

The participation in a benchmark study may be part of a benchmark provider offering. Benchmark study may be  
 10 defined as the collection and analysis of data for a certain customer group. Benchmark providers may offer this service to enable direct comparison of customer data against sample data but also to further increase the provider data base itself.

15

A further implementation of the inventive method comprises predefinable query assignment for selectable measures. An assignment of benchmark provider measures to standard measures will allow the customer to use  
 20 queries for easy data retrieval.

Step	Short description	Description and comments
1	Send measure List	Benchmark provider can notify customer what measures and benchmark data are required for their study.
2	Logon system	customer logon to system to retrieve data for measures

3	Create XML file	Scustomer reads data from his benchmark data storage for measures asked by provider and then generates a XML file for results.
4	Send XML file	Once XML file created customer can send it to provider via e-mail or ftp. Encryptions is highly recommended.
5	Analysis of data	Benchmark provider receives data and analysis of data. The result will be written into his own benchmark data base.

Get latest update on providers measure definition

- 5 The steps for this process are described in the following table with reference to Fig. 6.

Step	Short description	Description and comments
1	Logon	Customer logon to a financial service marketplace.
2	View pages	Under financial service page, customer can view e.g.: Benchmark provider list Benchmark provider service offerings Measure definitions of provider
3	Logon to system	Customer logon to system.

4	Upload latest measure definitions	Customer startup process from measure builder to upload latest measure definitions into customer measure catalog. All latest measure definitions are contained in a XML file. Upload process reads XML file from financial service marketplace.
---	-----------------------------------	---

#### Technical Aspects

- 5 The following paragraphs describes various technical design issues as they may apply to the benchmarking scenario.

Benchmark definition in measure builder:

- 10 The measure builder serves as the library for benchmark definition.

A measure may have the following attributes:

- a technical identifier (ID),
- a short text and/or long text,
- 15 • a formula or rule, how it is calculated or calculated from other measures,
- unlimited text description,
- a data source or reference to a data source,
- an assignment to categories, e.g. such as
- 20 industry or geography.

25



Measure builder features useful in this context include:

- organization of measure hierarchies where measures are linked based on functional areas such as HR, finance and logistic,
- link of measures to queries for data retrieval and reporting,
- benchmark data reporting based on the benchmark data storage,
- ability to apply filter to measures (Filter criteria may be industry, region etc.)

Measures may be grouped into a node within the measure catalog. Measures from benchmark providers may be grouped into dedicated nodes outside a standard measure hierarchy. Those provider measures may carry the same or a similar definition as standard measures. Provider measures may be pointed to standard measures. Benchmark provider measure ID's can be mapped to the technical ID's of measures in the measure builder.

By using data structures, systems and methods according to this invention, third benchmark provider may provide data for a subset of measures, but may also provide data for measures that are not yet part of the measure builder content or measures that have been added to the provider offering lately.

#### Benchmarking web pages

Benchmarking web page may provide access to provider information such as:

- basic description of service offering,
- URL link to navigate to provider website,

- 29 -

- "latest" measure definitions by individual provider.

Design and technical implementation of those services may be customized by the provider. Technical details  
5 may vary between the provider, such as how the XML files are delivered to customer, how the logon is handled etc and depend largely on solutions already in place and infrastructure considerations on the provider side. However, these details are within the knowledge  
10 of person of ordinary skill.

#### Benchmark exchange protocol

According to this invention, benchmark definitions and  
15 data may be exchanged via files in XML format. The XML scheme may be suggested by a customer or a software supplier according to the desired or available structure of the benchmark data storage and reviewed with the benchmark providers. Dimensions such as  
20 industry relevance, region, version, time characteristics may be reflected in the scheme.

#### Provider Web service

25 A provider's web page for exchanging the inventive data structures may offer services like:  
logon handling,  
enabling a customer to select specific or grouped benchmarks for download,  
30 generating valid XML files according to XML structure, filtering, sorting benchmarks,  
Documentation around benchmark service offering.

35

Import of benchmark data

In further implementations of the invention, means are provided for import functionality for:

validation of benchmark XML document,

5 parsing of imported file,

mapping of data to benchmark data storage,

viewing capability of imported data prior to posting to benchmark data storage,

posting of data to benchmark data storage,

10 monitoring and logging,

Error handling.

Those capabilities can be accomplished through various combinations of components like:

15 upload programs of economic software applications,

web application server,

business connector or other middleware functionality,

client-side applications (VB or Java),

browser capabilities, e.g. parsing.

20

Drill-down reporting

In case a first provider will capture only a subset of the data that is generally offered by a second

25 provider, .

Further drill-down and additional reporting

capabilities might be available on the provider's web page.

Therefore, a drill-down capability into the first

30 provider's web page that allows a customer to navigate to the analytical application of the second provider.

The realization of this capability may be handled in

several phases. In the first phase, a link to the

second provider web page with subsequent logon of the

35 user may be offered.

A dynamic link up to a report to report interface may be addressed in a second phase.

5

The following example provides an example of guidelines for implementing the inventive data structure. A limitation of the invention in any kind is not intended.

10

The inventive measure builder enables an enterprise, who uses this tool, for example to upload benchmarks and benchmark data and to compare benchmark data with their own company data.

15

A lots of benchmark providers in the market, do benchmark studies and send study result to their customers.

20

Any benchmark provider, who wants to provide benchmark data that a customer having a business software can upload directly into his system, can now according to the invention organize his benchmark data in a XML file that follows a structure as designed below.

25

Structure outline

Root element

The root element of the XML document is named as "benchMarkData".

30

Five parts may be included in the root element:

Memo, Provider, Meta Data, Measure ID and Transaction Data

35

#### Memo

Element memo is description for benchmark provider to add notes to this XML document.

5 Memo is optional element.

Meta Data type is defined by element mdCharacteristic, concrete meta data is defined by element mdCategory.

#### Provider

10 Element provider is used to define benchmark provider.

Each Benchmark Data Entry posted into business software system may be identified by benchmark provider.

15 Here is an example for provider element:

```
<provider id="FOO" name="Foo company"/>
```

20 Value of Attribute "id" identifies provider. Value of attribute "name" is description of provider, it's optional.

Another usage of provider's id is as prefix to provider

25 measure id, which is explained below in more detail.

#### Meta Data

Element metaData defines provider meta data, i.e.

30 classification standard, for example, industry category such as banking, chemical, education, etc.

It's optional for provider to include meta data in their Benchmark XML file.

35

Following pieces show how to define a meta data type as well as it's meta data:

```

5      <metaData>
      ...
      <mdCharacteristic id="OINDUSTRY" name="Industries"/>
      <mdCategory id="BANK" name="Banking"/>
      <mdCategory id="CHEM" name="Chemicals"/>
      ...
10     </mdCharacteristic>
      <mdCharacteristic id="OUNIT" name="Unit of measure"/>
      ...
      </metaData>

15     <metaData> begins definition of provider meta data.
      <mdCharacteristic id="OINDUSTRY" name="Industries"/>
      begins meta data entries for a specific meta data
      type. Value of Attribute "id" identifies meta data
20     type. Value of attribute "name" is for meta data type
      definition, it's optional. For valid meta data type
      refers to section 2.6.
      <mdCategory id="BANK" name="Banking"/> defines one
      meta data for the above given meta data type. Value
25     of attribute "id" gives technical name of meta data;
      value of attribute "name" is meta data description.
      <mdCategory id="CHEM" name="Chemicals"/> is another
      meta data for the above given meta data type.
      ...
30     </mdCharacteristic> ends definition of meta data for
      OINDUSTRY.
      </metaData> ends definition of provider meta data.

```

The meta data types are not random defined. The  
35 supported meta data types are listed below.

Measure ID (Identifier)

It's difficult for provider to define measure id in XML  
5 document.

If provider wants to use their own measure id in XML  
document, then the following steps are advisable:  
Send a document including a measure list to a third  
10 person who manages measure id.

Said third person give a unique name the benchmark  
provider as his identifier in element <provider>  
Provider's measure are created by adding the unique  
name as well as two separator'\', before and after  
15 the unique name, thus unique measure name is  
guaranteed

Publish provider measure definition onto a Service  
Marketplace

Customer download provider's measure definition into  
20 their business software system

Customer link provider measure to their company  
measure or

Customer activate provider measure as their company  
measure

25 When generating of benchmark value xml document,  
define provider's id as the above unique provider  
name

While reference to measure, just use the original  
measure id.

30 The reference to measure is defined by element  
<tdMeasure>.

If you want to refer to a original measure 1001, then  
define element like: <tdMeasure id="1001">

35

In some cases provider and customer may achieve agreement to use customer's measure id, for example, customer has one measure 'CUST\_0001', then refer to this measure like:

5     <tdMeasure id="CUST\_1001" original="N">

Provider's id may be defined as agreed unique name in the second step if it is required to use the original measure id in the xml document.

10

Name restriction to provider's original measure id may be:

Only alphanumeric and '\_' are allowed in measure name, length may be less than 11.

15

Transaction Data

Element transactionData contains benchmark data entries to be posted into customer system.

20

It's advantageous to have at least one entry in transaction data, or else it doesn't make sense to delivery such XML file to customer.

25

Following pieces show exemplary how to define benchmark data for a measure turn over rate (measure id 1001) :

<transactionData>

30

    <tdMeasure id="1001">

        <tdCharacteristic id="OCALYEAR">

            <tdCategory id="1999">

                <tdCharacteristic id="OBM\_S\_TYPE">

                    <tdCategory id="GRO">

35

                        <tdCharacteristic id="OBM\_S\_RANGE">

- 36 -



```

        <tdCategory id="H">
            <tdValue>
                <tdValueItem
5         id="25%">0.159</tdValueItem>
                <tdValueItem
            id="MED">0.22</tdValueItem>
                <tdValueItem
            id="75%">0.38</tdValueItem>
                <unit>%</unit>
10        </tdValue>
        </tdCategory>
        <tdCategory id="M">
            ...
        ...
15    </tdMeasure>
        ...
    </transactionData>

20    <transactionData> begins definition for benchmark
        data entries.
    <tdMeasure id="1001"> begins benchmark data entries
        for measure \FOO\1001
25    <tdCharacteristic id="OCALYEAR"> together with
        <tdCategory id="1999"> defines header meta data
        calendar year :1999. For valid meta data type refers
        to section 2.6
    <tdCharacteristic id="OBM_S_TYPE"> together with
30    <tdCategory id="GRO"> defines header meta data
        company size type: revenue growths.
    <tdCharacteristic id="OBM_S_RANGE"> together with
    <tdCategory id="H"> defines header meta data company
        size range: high revenue growths.

```

<tdValue> begins definition of entry items. And up to now we have header data like below:

Measure ID	Calendar Year	Company size type	Company size range
OHR_HRMC_006	1999	GRO	H

- 5 <tdValueItem id="25%">0.159</tdValueItem> benchmark value 0.159 for benchmark version "25%".  
 <tdValueItem id="MED">0.22</tdValueItem> benchmark value 0.22 for benchmark version "MED".  
 <tdValueItem id="75%">0.38</tdValueItem> benchmark  
 10 value 0.38 for benchmark version "75%".  
 <unit>%</unit> benchmark value unit is %.  
 </tdValue> ends definition of entry items. Up to now we have three entries like below:

Measure ID	Calendar Year	Company size type	Company size range	Benchmark Version	Unit	Value
OHR_HRMC_006	1999	GRO	H	25%	%	0.159
OHR_HRMC_006	1999	GRO	H	MED	%	0.22
OHR_HRMC_006	1999	GRO	H	75%	%	0.38

- 15 </tdCategory>  
 <tdCategory id="M"> begins definition of benchmark data entries that have the same measure id, calendar year, company size type but with different company size range.  
 20 ...  
 </tdMeasure> ends definition for benchmark data entries for measure OHR\_HRMC\_006.  
 ...  
 </transactionData> ends definition for  
 25 benchmark data entries

Any number of meta data types as many as are needed can be defined for benchmark data entries.

#### Meta data types

5

In the table below meta data types are listed.

Table 1: meta data types

Meta data type id	Description	Examples
OMEASURE	Measure	ROCE;NOPAT;FTE
OBM_VERSION	Benchmark Version	25% percentile; best in class
OINDUSTRY	Industry	Banking; Education
OCOUNTRY	Country	Germany; France
OBM_PROVIDE	Benchmark Value Provider	InfoHRM;PMG;Self
OBM_REGION	Geographic region	Middle Europe; East North Asia
OBM_COMPANY	Company	X;Y;Z
OBM_S_TYPE	Company size type	Revenue;Employee numbers
OBM_S_RANGE	Company size range	5m-10m;1000-5000
OUNIT	Unit for measure quantity	%;Ratio
OCURRENCY	Currency for measure amount	USD;DEM;EURO
OCALMONTH2	Calendar month with two digits	01;02;11
OCALMONTH	Calendar Year / Month	200001; 200112
OCALQUART1	Calendar quarter with one digit	1;2;3;4
OCALQUARTER	Calendar Year/	200101; 200102;

- 39 -

Meta data type id	Description	Examples
	Quarter	200103;200104
OCALYEAR	Calendar year with four digits	2000;2001
OFISCPER	Fiscal year period	2000001;2001002
OFISCPER3	Fiscal year period	001;002
OFISCYEAR	Fiscal Year	1999;2000;2001
OFISCVARNT	Fiscal Year Variant	K4
OBM_VALID_F	Benchmark Valid Date From	20010101
OBM_VALID_T	Benchmark Valid Date To	20011231

Provider is able to define meta data for all listed meta data types except for the first one OMEASURE as measure can be defined via Measure Builder.

5

When defining data entry for benchmark values, meta data may be referenced by different ways:

1. Measures are identified by attribute "id" of element tdMeasure;
- 10 2. Benchmark provider is identified by element provider.
3. Units are identified by element unit;
4. Currencies are identified by element currency;
5. Benchmark versions are identified by attribute "id" of element tdValueItem;
- 15 6. For other meta data types, meta data are identified by attribute "id" of both element tdCharacteristic and tdCategory.
- 20 Other meta data types may be added.

## Proposed XML schema

```

5  <?xml version="1.0" encoding="UTF-8"?>
    <xsd:schema
      xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
      elementFormDefault="qualified">
      <xsd:element name="benchMarkData">
        <xsd:complexType>
10         <xsd:sequence>
            <xsd:element ref="memo" minOccurs="0"/>
            <xsd:element ref="provider"/>
            <xsd:element ref="metaData"
15             minOccurs="0"/>
            <xsd:element ref="transactionData"/>
          </xsd:sequence>
          <xsd:attribute name="version" use="default"
            value="1.0">
            <xsd:simpleType>
20             <xsd:restriction
              base="xsd:string"/>
            </xsd:simpleType>
          </xsd:attribute>
        </xsd:complexType>
25      </xsd:element>
      <xsd:element name="memo">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:maxLength value="255"/>
30          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="provider">
        <xsd:complexType>
          <xsd:attribute name="id" use="required">
35          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:maxLength value="60"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:complexType>
      </xsd:element>
    </xsd:schema>

```

- 41 -

```

        </xsd:restriction>
    </xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="name" use="optional">
5    <xsd:simpleType>
        <xsd:restriction base="xsd:string">
            <xsd:maxLength value="60"/>
        </xsd:restriction>
    </xsd:simpleType>
10    </xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="metaData">
    <xsd:complexType>
15    <xsd:sequence>
        <xsd:element ref="mdCharacteristic"
maxOccurs="unbounded"/>
    </xsd:sequence>
    </xsd:complexType>
20 </xsd:element>
<xsd:element name="transactionData">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element ref="tdMeasure"
25 maxOccurs="unbounded"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="mdCharacteristic">
30 <xsd:complexType>
    <xsd:sequence>
        <xsd:element ref="mdCategory"
maxOccurs="unbounded"/>
    </xsd:sequence>
35 <xsd:attribute name="id" use="required">
    <xsd:simpleType>
        <xsd:restriction base="xsd:string"/>
    </xsd:simpleType>

```

```

        </xsd:attribute>
        <xsd:attribute name="name" use="optional">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
5              <xsd:maxLength value="60"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:complexType>
10    </xsd:element>
    <xsd:element name="mdCategory">
      <xsd:complexType>
        <xsd:attribute name="id" use="required">
          <xsd:simpleType>
15            <xsd:restriction base="xsd:string">
              <xsd:maxLength value="60"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:attribute>
20        <xsd:attribute name="name" use="optional">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:maxLength value="60"/>
            </xsd:restriction>
25          </xsd:simpleType>
        </xsd:attribute>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="tdMeasure">
30      <xsd:complexType>
        <xsd:sequence>
          <xsd:element ref="tdCharacteristic"
maxOccurs="unbounded"/>
        </xsd:sequence>
35      <xsd:attribute name="id" use="required">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:maxLength value="16"/>

```

```

        </xsd:restriction>
        </xsd:simpleType>
        </xsd:attribute>
        <xsd:attribute name="original"
5   type="xsd:boolean" use="default" value="Y" />
        </xsd:complexType>
        </xsd:element>
        <xsd:element name="tdCharacteristic">
        <xsd:complexType>
10   <xsd:sequence>
        <xsd:element ref="tdCategory"
maxOccurs="unbounded"/>
        </xsd:sequence>
        <xsd:attribute name="id" use="required">
15   <xsd:simpleType>
        <xsd:restriction base="xsd:string"/>
        </xsd:simpleType>
        </xsd:attribute>
        </xsd:complexType>
20   </xsd:element>
        <xsd:element name="tdCategory">
        <xsd:complexType>
        <xsd:choice>
        <xsd:element ref="tdCharacteristic"
25   maxOccurs="unbounded"/>
        <xsd:element ref="tdValue"/>
        </xsd:choice>
        <xsd:attribute name="id" use="required">
        <xsd:simpleType>
30   <xsd:restriction base="xsd:string">
        <xsd:maxLength value="60"/>
        </xsd:restriction>
        </xsd:simpleType>
        </xsd:attribute>
35   </xsd:complexType>
        </xsd:element>
        <xsd:element name="tdValue">
        <xsd:complexType>

```



```

        <xsd:sequence>
            <xsd:element ref="tdValueItem"
maxOccurs="unbounded"/>
            <xsd:choice>
5                <xsd:element ref="unit"/>
                <xsd:element ref="currency"/>
            </xsd:choice>
            <xsd:element ref="sample" minOccurs="0"/>
            <xsd:element ref="sampleDateFrom" minOccurs="0"/>
10            <xsd:element ref="sampleDateTo" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="tdValueItem">
15    <xsd:complexType>
        <xsd:simpleContent>
            <xsd:extension base="xsd:float">
                <xsd:attribute name="id" use="required">
                    <xsd:simpleType>
20                        <xsd:restriction base="xsd:string">
                            <xsd:maxLength value="60"/>
                        </xsd:restriction>
                    </xsd:simpleType>
                </xsd:attribute>
25            </xsd:extension>
        </xsd:simpleContent>
    </xsd:complexType>
</xsd:element>
<xsd:element name="sample" type="xsd:decimal"/>
30 <xsd:element name="sampleDateFrom" type="xsd:date"/>
<xsd:element name="sampleDateTo" type="xsd:date"/>
<xsd:element name="unit">
    <xsd:simpleType>
        <xsd:restriction base="xsd:string">
35            <xsd:maxLength value="60"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>

```

```

    <xsd:element name="currency">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:maxLength value="60"/>
5      </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
    </xsd:schema>

```

#### 10 Class diagram for XML schema

The illustration in Fig. 7 shows a relation among different elements.

#### 15 Sample XML file for the proposal

```

<?xml version="1.0"?>
<benchMarkData version="1.0"
xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instance"
20 xsi:noNamespaceSchemaLocation="BenchmarkData.xsd">
  <memo>This package contains HR BenchmarkData.</memo>
  <provider id="FOO" name="Foo company"/>
  <metaData>
    <mdCharacteristic id="0UNIT" name="Unit">
25    <mdCategory id="%" name="Percentages"/>
    </mdCharacteristic>
    <mdCharacteristic id="0CURRENCY" name="Currency">
    <mdCategory id="USD" name="American Dollars"/>
    </mdCharacteristic>
30    <mdCharacteristic id="0BM_S_TYPE" name="Company Size
Types">
    <mdCategory id="001" name="Revenue Growths"/>
    <mdCategory id="002" name="Number of Employees"/>
    </mdCharacteristic>
35    <mdCharacteristic id="0BM_S_RANGE" name="Comapny Size
Ranges">
    <mdCategory id="HIGH" name="High Revenue Growths"/>
    <mdCategory id="MED" name="Medium Revenue Growths"/>

```

```

        <mdCategory id="LOW" name="Low Revenue Growths"/>
        <mdCategory id="500" name="1 - 500 employees"/>
        <mdCategory id="1000" name="501 - 1,000 employees"/>
        <mdCategory id="2000" name="1,001 - 2,000
5 employees"/>
        <mdCategory id="2000+" name="more than 2000+
employees"/>
    </mdCharacteristic>
    <mdCharacteristic id="0INDUSTRY" name="Industries">
10    <mdCategory id="ALL" name="All industries"/>
    <mdCategory id="BANK" name="Banking"/>
    <mdCategory id="CHEM" name="Chemicals"/>
    </mdCharacteristic>
    <mdCharacteristic id="0BM_VERSION" name="Benchmark
15 versions">
        <mdCategory id="25%" name="25th Percentile"/>
        <mdCategory id="MED" name="Median"/>
        <mdCategory id="75%" name="75th Percentile"/>
    </mdCharacteristic>
20 </metaData>
    <transactionData>
        <tdMeasure id="1001">
            <tdCharacteristic id="0CALYEAR">
                <tdCategory id="1999">
25    <tdCharacteristic id="0BM_S_TYPE">
                <tdCategory id="001">
                    <tdCharacteristic id="0BM_S_RANGE">
                        <tdCategory id="HIGH">
                            <tdValue>
30    <tdValueItem
id="25%">0.159</tdValueItem>
                            <tdValueItem
id="MED">0.22</tdValueItem>
                            <tdValueItem
35 id="75%">0.38</tdValueItem>
                                <unit>%</unit>
                            </tdValue>
                        </tdCategory>

```

```

                                <tdCategory id="MED">
                                    <tdValue>
                                        <tdValueItem
5      id="25%">0.111</tdValueItem>
                                        <tdValueItem
      id="MED">0.165</tdValueItem>
                                        <tdValueItem
      id="75%">0.252</tdValueItem>
                                        <unit>%</unit>
10     </tdValue>
      </tdCategory>
    </tdCharacteristic>
  </tdCategory>
  <tdCategory id="002">
15    <tdCharacteristic id="OBM_S_RANGE">
      <tdCategory id="500">
          <tdValue>
              <tdValueItem
      id="25%">0.159</tdValueItem>
20          <tdValueItem
      id="MED">0.1825</tdValueItem>
              <tdValueItem
      id="75%">0.2535</tdValueItem>
              <unit>%</unit>
25          </tdValue>
      </tdCategory>
      <tdCategory id="1000">
          <tdValue>
              <tdValueItem
30      id="25%">0.113</tdValueItem>
              <tdValueItem
      id="MED">0.169</tdValueItem>
              <tdValueItem
      id="75%">0.302</tdValueItem>
35          <unit>%</unit>
          </tdValue>
      </tdCategory>
    </tdCharacteristic>

```

```

        </tdCategory>
      </tdCharacteristic>

      <tdCharacteristic id="0INDUSTRY">

        <tdCategory id="BANK">

          <tdValue>
            <tdValueItem
              id="25%">0.217</tdValueItem>
            <tdValueItem
              id="MED">0.279</tdValueItem>
            <tdValueItem
              id="75%">0.333</tdValueItem>
            <unit>%</unit>
          </tdValue>
        </tdCategory>
      <tdCategory id="CHEM">
        <tdValue>
          <tdValueItem
            id="25%">0.054</tdValueItem>
          <tdValueItem
            id="MED">0.146</tdValueItem>
          <tdValueItem
            id="75%">0.172</tdValueItem>
          <unit>%</unit>
        </tdValue>
      </tdCategory>
    </tdCharacteristic>
  </tdCategory>
</tdCharacteristic>
</tdMeasure>
<tdMeasure id="CUST_0002" original="N">
  <tdCharacteristic id="0CALYEAR">

```

```

        <tdCategory id="1999">
            <tdCharacteristic id="OBM_S_TYPE">
                <tdCategory id="001">
                    <tdCharacteristic id="OBM_S_RANGE">
8                     <tdCategory id="HIGH">
                            <tdValue>
                                <tdValueItem
id="25%">0.159</tdValueItem>
                                <tdValueItem
10 id="MED">0.22</tdValueItem>
                                    <tdValueItem
id="75%">0.38</tdValueItem>
                                <unit>%</unit>
                            </tdValue>
15                        </tdCategory>
                            <tdCategory id="MED">
                                <tdValue>
                                    <tdValueItem
id="25%">0.111</tdValueItem>
20                                <tdValueItem
id="MED">0.165</tdValueItem>
                                    <tdValueItem
id="75%">0.252</tdValueItem>
                                <unit>%</unit>
25                        </tdValue>
                    </tdCategory>
                </tdCharacteristic>
            </tdCategory>
            <tdCategory id="002">
30                <tdCharacteristic id="OBM_S_RANGE">
                    <tdCategory id="500">
                        <tdValue>
                            <tdValueItem
id="25%">0.159</tdValueItem>
35                            <tdValueItem
id="MED">0.1825</tdValueItem>
                                <tdValueItem
id="75%">0.2535</tdValueItem>

```

- 50 -

```

                                <unit>%</unit>
                                </tdValue>
                                </tdCategory>
                                <tdCategory id="1000">
5                                <tdValue>
                                <tdValueItem
                                id="25%">0.113</tdValueItem>
                                <tdValueItem
                                id="MED">0.169</tdValueItem>
10                                <tdValueItem
                                id="75%">0.302</tdValueItem>
                                <unit>%</unit>
                                </tdValue>
                                </tdCategory>
15                                </tdCharacteristic>
                                </tdCategory>
                                </tdCharacteristic>
                                <tdCharacteristic id="0INDUSTRY">
                                <tdCategory id="BANK">
20                                <tdValue>
                                <tdValueItem id="25%">0.217</tdValueItem>
                                <tdValueItem id="MED">0.279</tdValueItem>
                                <tdValueItem id="75%">0.333</tdValueItem>
                                <unit>%</unit>
25                                </tdValue>
                                </tdCategory>
                                <tdCategory id="CHEM">
                                <tdValue>
                                <tdValueItem id="25%">0.054</tdValueItem>
                                <tdValueItem id="MED">0.146</tdValueItem>
30                                <tdValueItem id="75%">0.172</tdValueItem>
                                <unit>%</unit>
                                </tdValue>
                                </tdCategory>
35                                </tdCharacteristic>
                                </tdCategory>
                                </tdCharacteristic>
                                </tdMeasure>

```

< /transactionData >

< /benchMarkData >

Modifications and adaptations of the present invention will be  
5 apparent to those skilled in the art from consideration of the  
specification and practice of the invention disclosed herein. The  
foregoing description of an implementation of the invention has been  
presented for purposes of illustration and description.

It is not exhaustive and does not limit the invention to the precise  
10 form disclosed. Modifications and variations are possible in light of  
the above teachings or may be acquired from the practicing of the  
invention. For example, the described implementation includes  
software, but systems and methods consistent with the present  
invention may be implemented as a combination of hardware and  
15 software or in hardware alone. Additionally, although aspects of the  
present invention are described for being stored in memory, one  
skilled in the art will appreciate that these aspects can also be stored  
on other types of computer-readable media, such as secondary  
storage devices, for example, hard disks, floppy disks, or CD-ROM;  
20 the Internet or other propagation medium; or other forms of RAM or  
ROM. It is intended that the specification and examples be  
considered as exemplary only, with a true scope and spirit of the  
invention being indicated by the following claims.

25 Computer programs based on the written description and flow charts  
of this invention are within the skill of an experienced developer.

A reference herein to a prior art document is not an admission that  
the document forms part of the common general knowledge in the art  
30 in Australia.



- 52a -

- In the claims which follow and in the preceding description of the invention, except where the context requires otherwise due to express language or necessary implication, the word “comprise” or variations such as “comprises” or “comprising” is used in an inclusive sense,
- 5 i.e. to specify the presence of the stated features but not to preclude the presence or addition of further features in various embodiments of the invention.

2003206891 17 Apr 2008

53

What is claimed is:

1. A method for processing electronic data structures, the electronic data structures each comprising a benchmark definition including rules for calculating and interpreting benchmark data, and having a file format that is readable by a SGML-based language, the method comprising the steps of:
  - receiving one or more of said electronic data structures by electronic means; and
  - mapping a benchmark definition contained in said electronic data structure to equivalent benchmark definition contained in software for supporting business processes by using a table, in which a first ID of the benchmark definition in the electronic data structure is assigned to a second ID of the benchmark definition in the software for the support of business processes.
2. The method of claim 1, wherein the electronic means comprises the Internet and wherein a link to said one or more of the electronic data structures is presented on an Internet page, which link is used to receive one or more of the electronic data structures.
3. The method of claim 1, further comprising:
  - receiving one or more of said data structures by email or SMS.
4. The method of one or more of claim 1, further comprising:
  - sending one or more of said electronic data structures by using at least one of an http and an ftp protocol.

- 54 -

2003206891 17 Apr 2008

5. The method of claim 4, further comprising:  
incorporating the benchmark definition contained in said one or  
more electronic data structures into software for supporting business  
processes.
- 5 6. The method of claim 5, wherein the incorporation of the benchmark  
definition is performed automatically, and further wherein the method  
comprises:  
reading one or more of said electronic data structures;  
10 recognising one or more key words for the benchmark definition  
within one or more of said electronic data structures; and  
writing information, identified by one or more of said key words,  
into a data base structure.
- 15 7. The method of claim 6, wherein the information, identified by a key  
word, and the data base structure, into which said information is written,  
are assigned to each other by an algorithm.
- 20 8. The method of claim 7, wherein said algorithm comprises one or  
more tables or one or more rules.
9. The method of claim 8, wherein said one or more rules are contained  
in one or more tables.
- 25 10. The method of claim 9, wherein said data base structure is  
predefinable.

N:\Sydney\Cases\Patent\53000-53999\F53005.AU\Spec\IP53085.AU Specification 2008-4-10.doc 17/04/08

- 55 -

2003206891 17 Apr 2008

11. The method of claim 1, wherein the method is for use in software for supporting business processes, and further wherein the software comprises enterprise resource planning software.

- 5 12. A computer system for processing electronic data structures, the electronic data structures comprising a benchmark definition including rules for calculating and interpreting benchmark data, and having a file format that is readable by a SGML-based language, the system comprising:
- 10 memory means having program instructions;  
input means for entering data;  
storage means for storing data;  
a processor responsive to the program instructions for receiving one or more of said electronic data structures by electronic means; and  
means for mapping the benchmark definition contained in said
- 15 electronic data structure to an equivalent benchmark definition contained in software for support of business processes by using a table, in which a first ID of the benchmark definition in the electronic data structure is assigned to a second ID of the benchmark definition in the software for the support of business processes.

- 20 13. The computer system of claim 12, wherein the electronic means comprises the Internet and wherein a link to said one or more of the electronic data structures is presented on an Internet page, which link is used to receive one or more of the electronic data structures.

- 25 14. The computer system of claim 12, further comprising:  
means for receiving one or more of said data structures by email or SMS.

M:\Sydney\Cases\Patent\68000-68999\655065.AU\Spec\9P63085.AU Specification 2008-4-16.doc 17/04/08

- 56 -

2003206891 17 Apr 2008

15. The computer system of claim 12, further comprising:  
means for sending one or more of said electronic data structures by  
using at least one of an http or an ftp protocol.
- 5 16. The computer system of claim 15, wherein the system incorporates  
the benchmark definition contained in said one or more electronic data  
structures into software for supporting business processes.
- 10 17. The computer system of claim 16, wherein the incorporation of the  
benchmark definition is performed automatically, and wherein the system  
further comprises:  
means for reading one or more of said electronic data structures;  
means for recognising one or more key words for benchmark  
definition within one or more of said electronic data structures; and  
15 means for writing information, identified by one or more of said key  
words, into a data base structure.
18. The computer system of claim 17, wherein the information,  
identified by a key word, and the data base structure, into which said  
20 information is written, are assigned to each other by an algorithm.
19. The computer system of claim 18, wherein said algorithm comprises  
one or more tables or one or more rules.
- 25 20. The computer system of claim 19, wherein said one or more rules are  
contained in one or more tables.
21. The computer system of one or more of claim 17, wherein said data  
base structure is predefinable.

I:\Sydney\Cases\Patient\63000-63999\PS3085.AU\Specs\PS3085.AU Specification 2008-4-10.doc 17/04/08

- 57 -

2003206891 17 Apr 2008

22. The computer system of claim 15, wherein the system is for use in a software for supporting business processes, the software comprising enterprise resource planning software.
- 5 23. A computer program product for processing electronic data structures, the electronic data structures comprising a benchmark definition including rules for calculating and interpreting benchmark data, and having a file format that is readable by a SGML-based language, the program comprising at least one instruction which, when implemented on a
- 10 computer readable medium of a computing system, causes the computing system to implement the method steps according to any one of claims 1 to 11.
24. A computer readable medium providing a computer program product
- 15 in accordance with claim 23.
25. A computer data signal embodied in a computer readable medium comprising code for processing electronic data structures, the electronic data structures each comprising a benchmark definition and having a file
- 20 format that is readable by a SGML-based language, said code comprising instructions for implementing the method according to any one of claims 1 to 11.

H:\Sydney\Cases\Poten\53000-53999\P53005.AU\Spc\IP53005.AU Specification 2008-4-10.doc 17/04/08

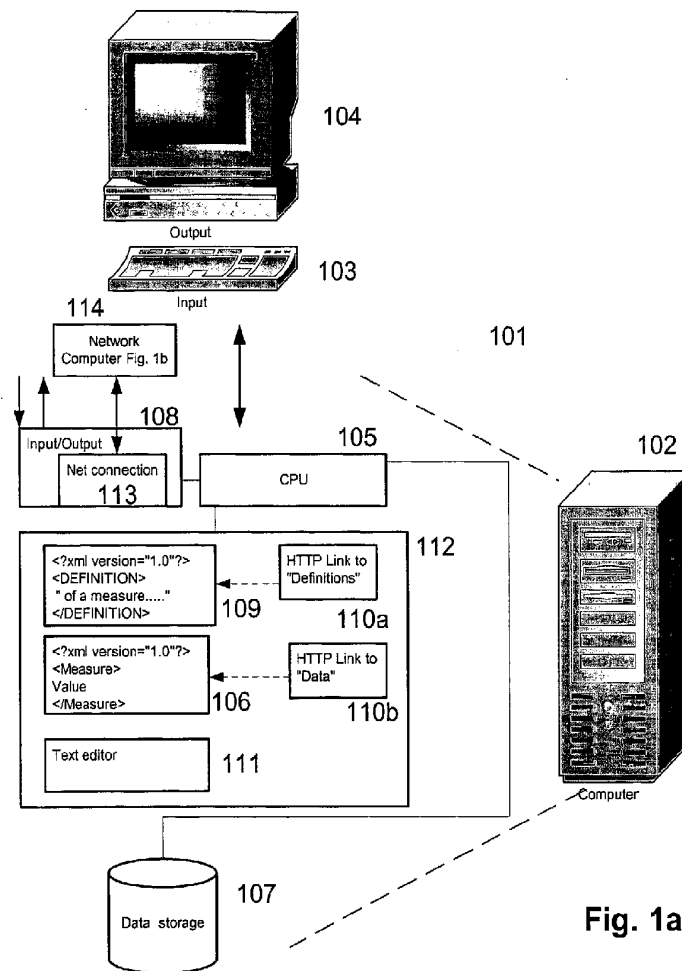


Fig. 1a

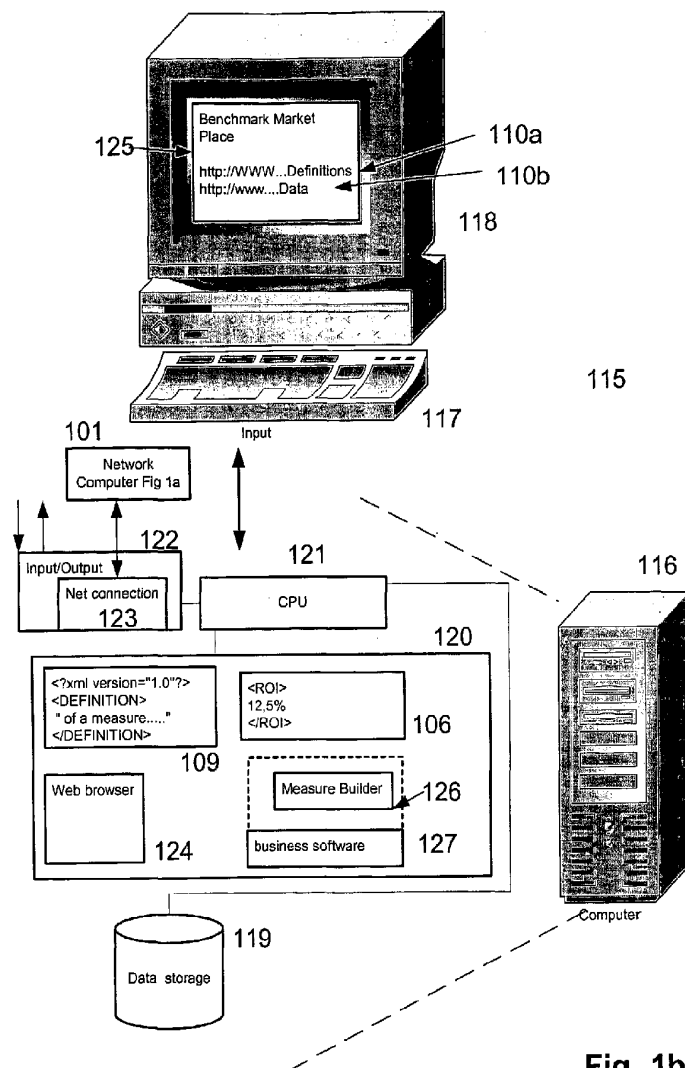


Fig. 1b



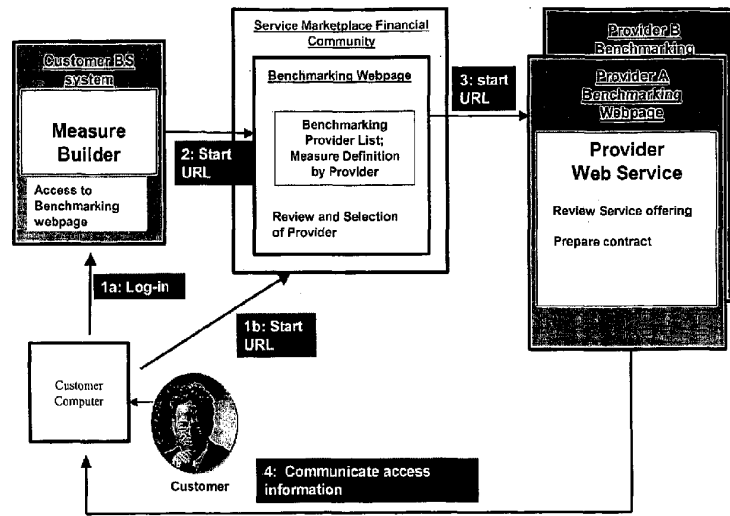


Fig. 2

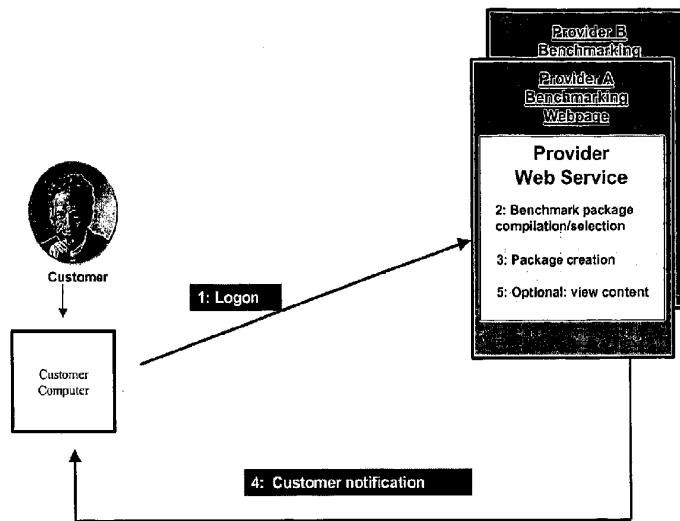


Fig. 3

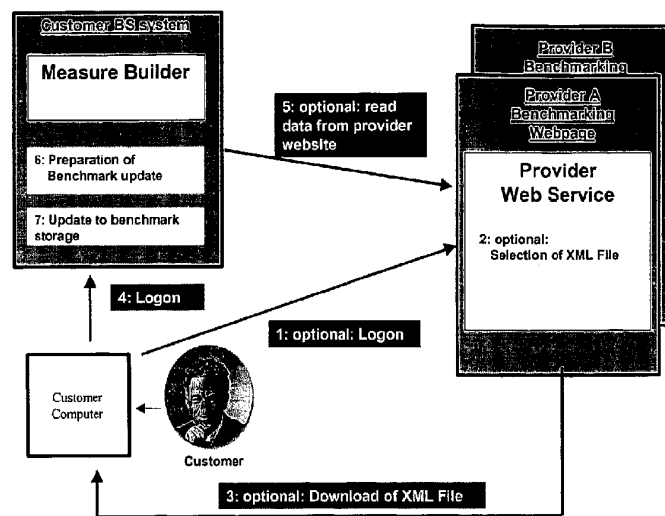


Fig. 4

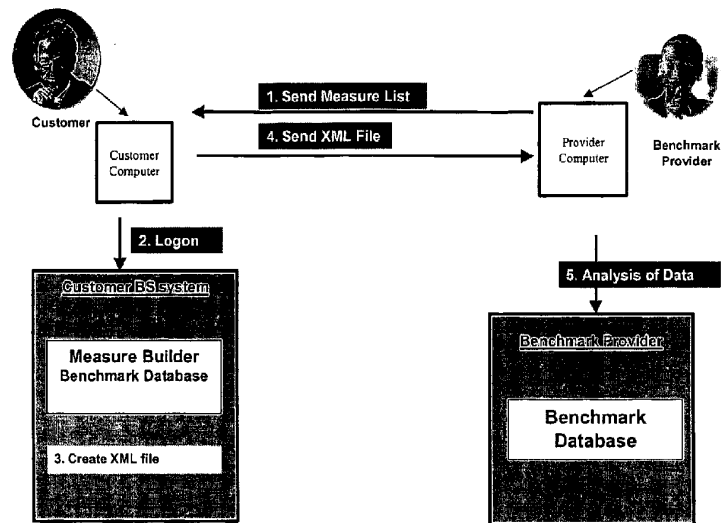


Fig. 5

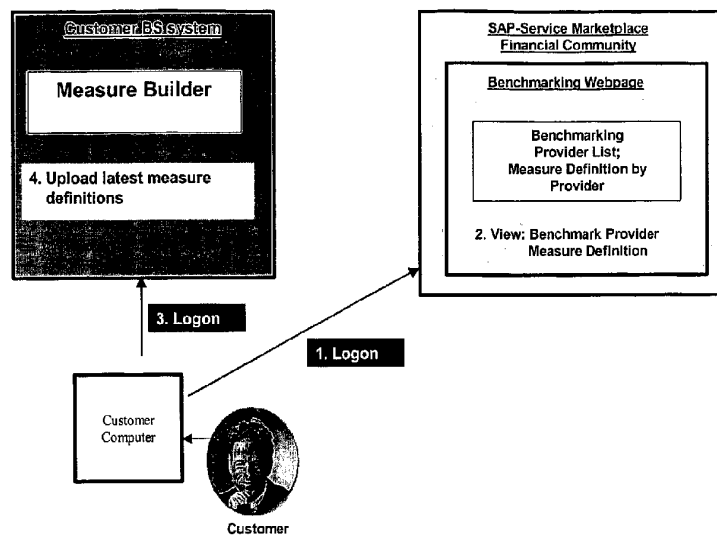


Fig. 6

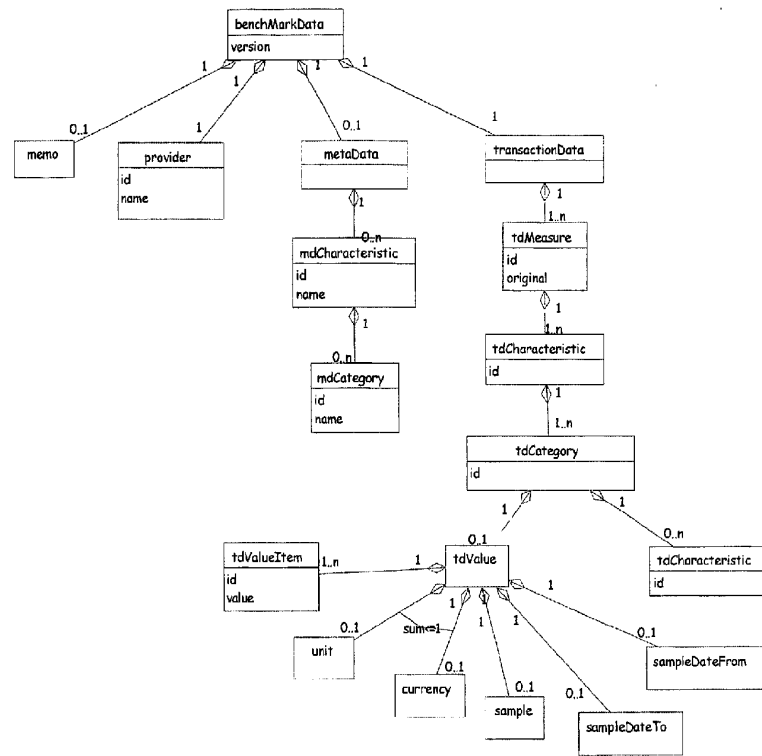


Fig. 7