



US 20060143115A1

(19) **United States**

(12) **Patent Application Publication**

Eder

(10) **Pub. No.: US 2006/0143115 A1**

(43) **Pub. Date: Jun. 29, 2006**

(54) **ENTERPRISE RISK MANAGEMENT SYSTEM**

Related U.S. Application Data

(63) Continuation of application No. 10/329,172, filed on Dec. 23, 2002, which is a continuation-in-part of application No. 09/688,983, filed on Oct. 17, 2000.

(76) Inventor: **Jeffrey Scott Eder**, Mill Creek, WA (US)

Publication Classification

Correspondence Address:
Jeff Eder
19108 30th Drive SE
Mill Creek, WA 98012 (US)

(51) **Int. Cl.**
G06Q 40/00 (2006.01)
(52) **U.S. Cl.** 705/38

(57) **ABSTRACT**

(21) Appl. No.: **11/360,087**

A method and system (100) for identifying, measuring and managing the risk associated with a commercial enterprise.

(22) Filed: **Feb. 23, 2006**

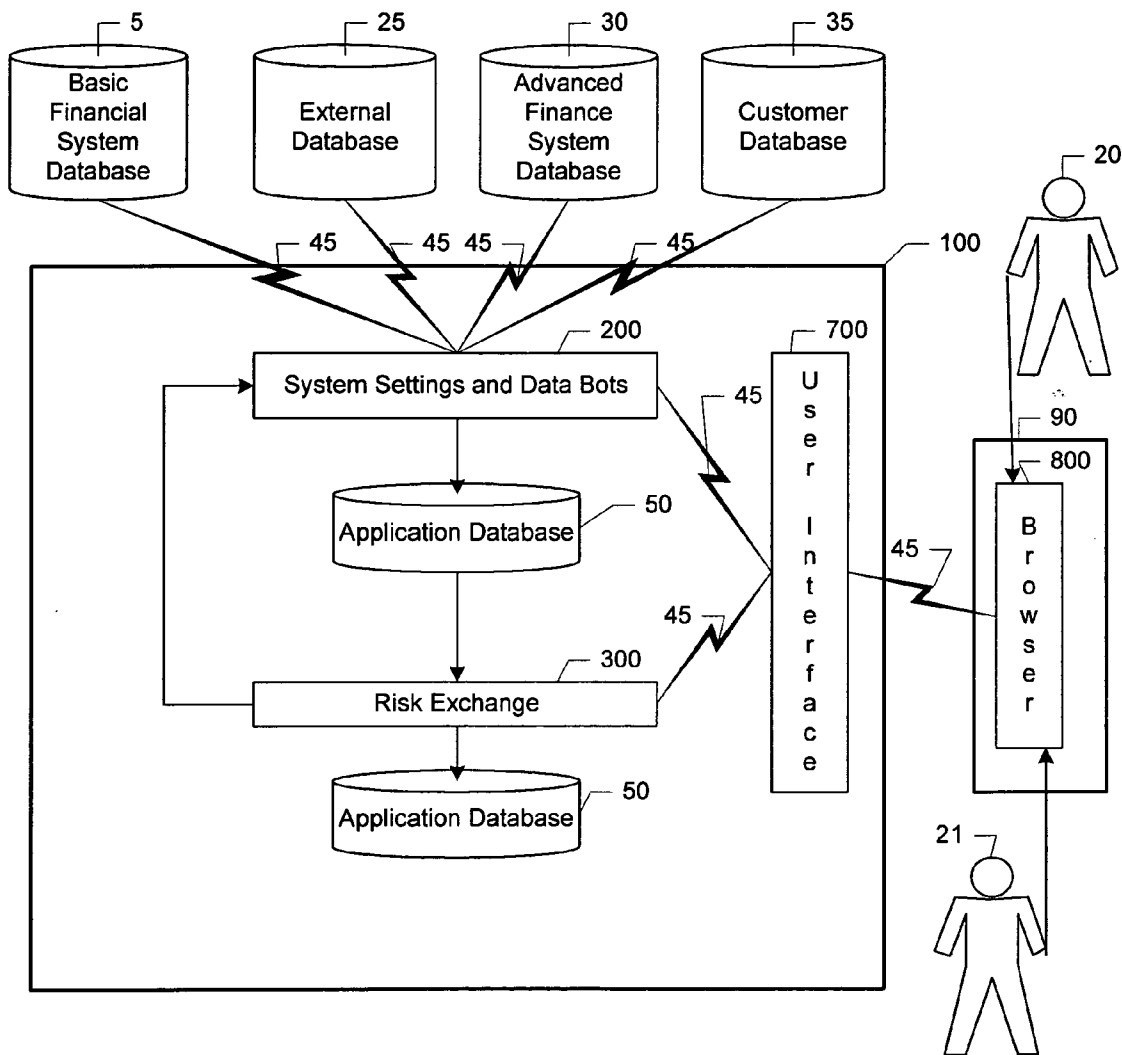


FIG. 1

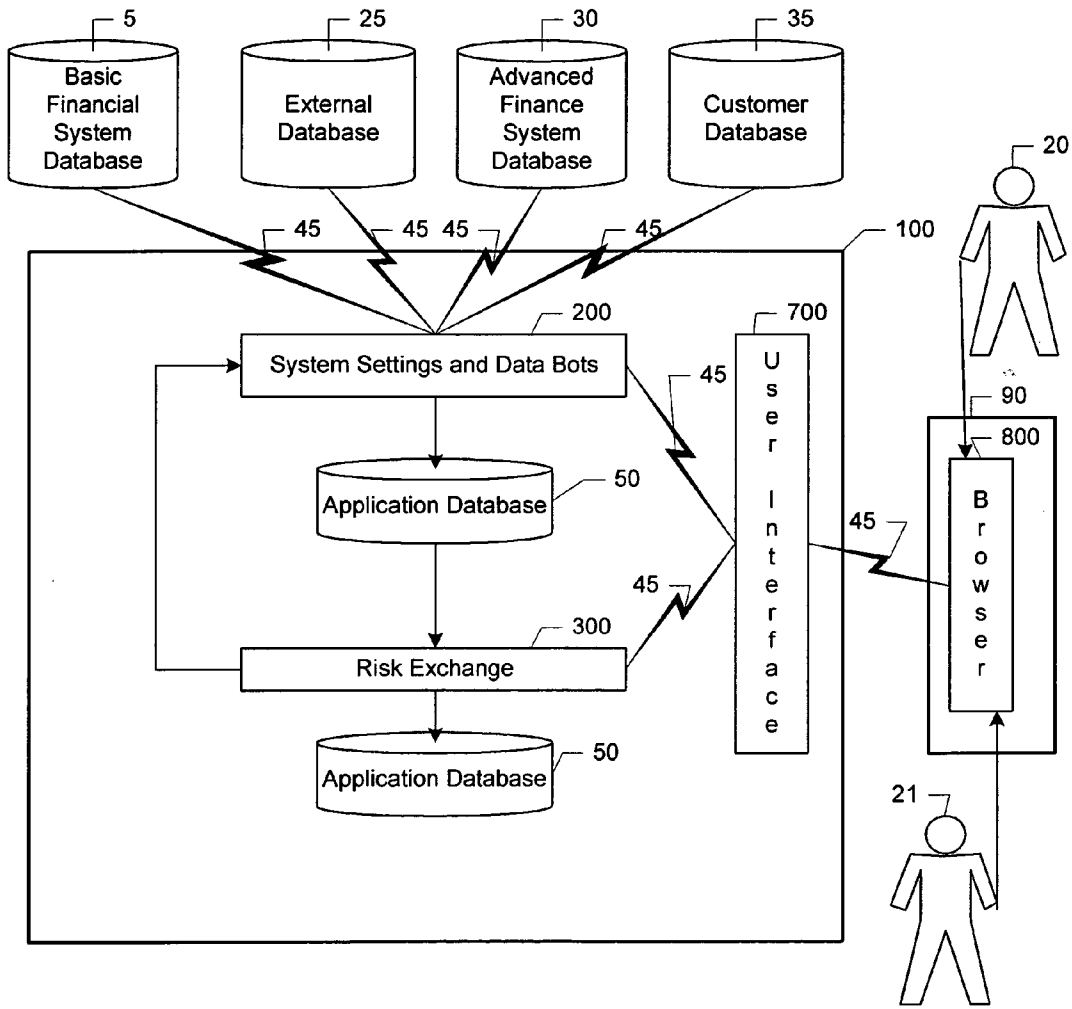


FIG. 2

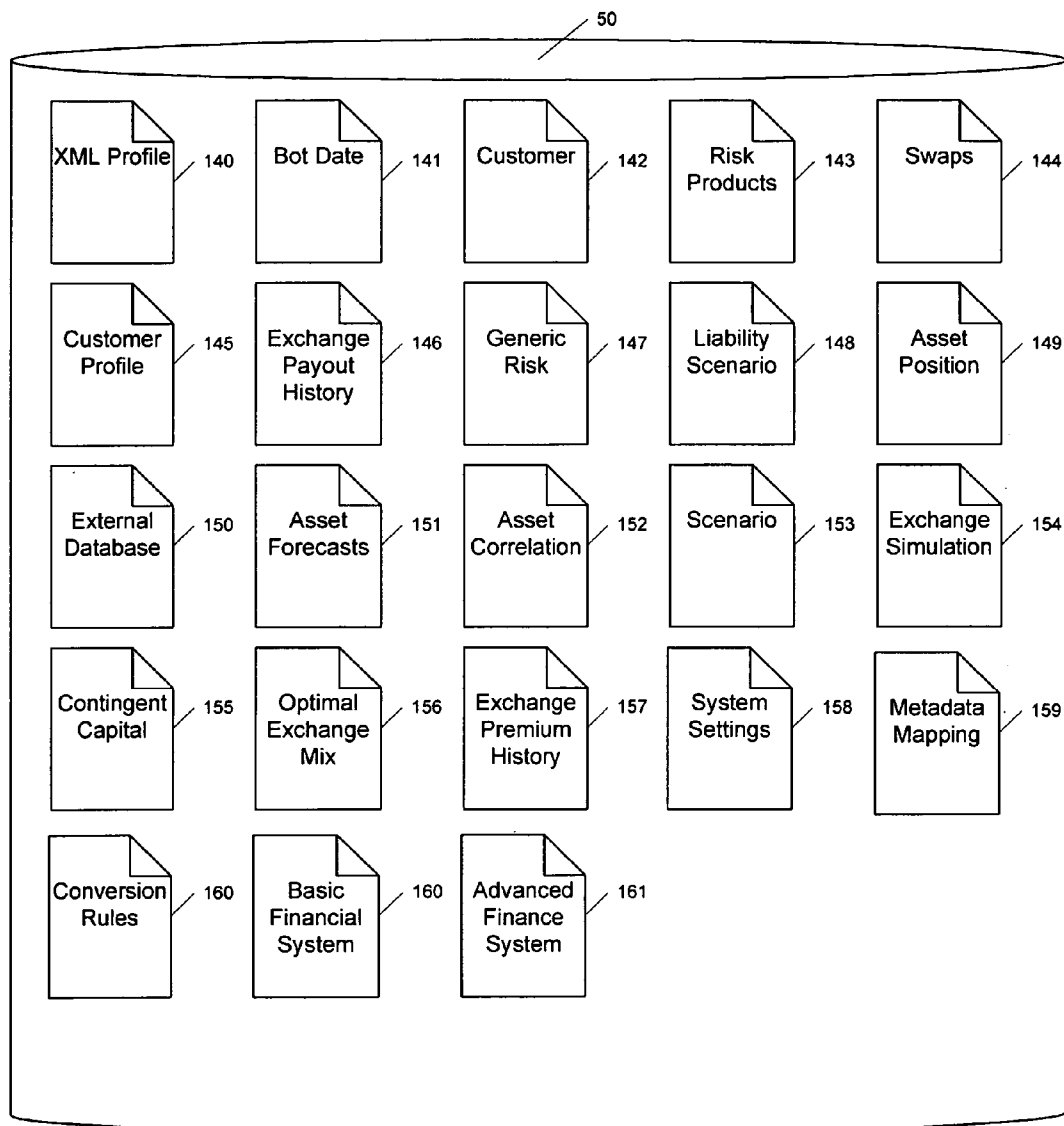


FIG. 3

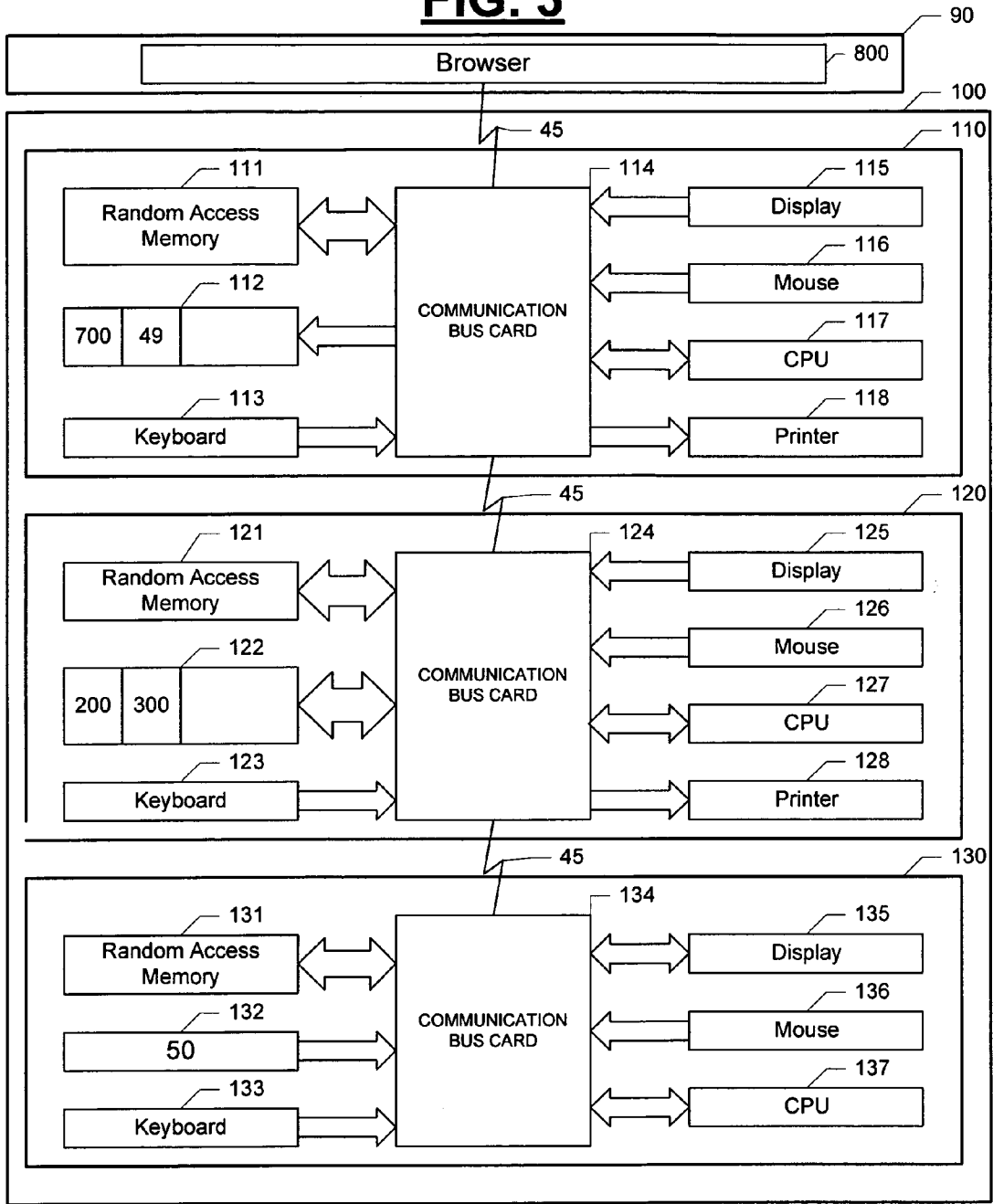


FIG. 4

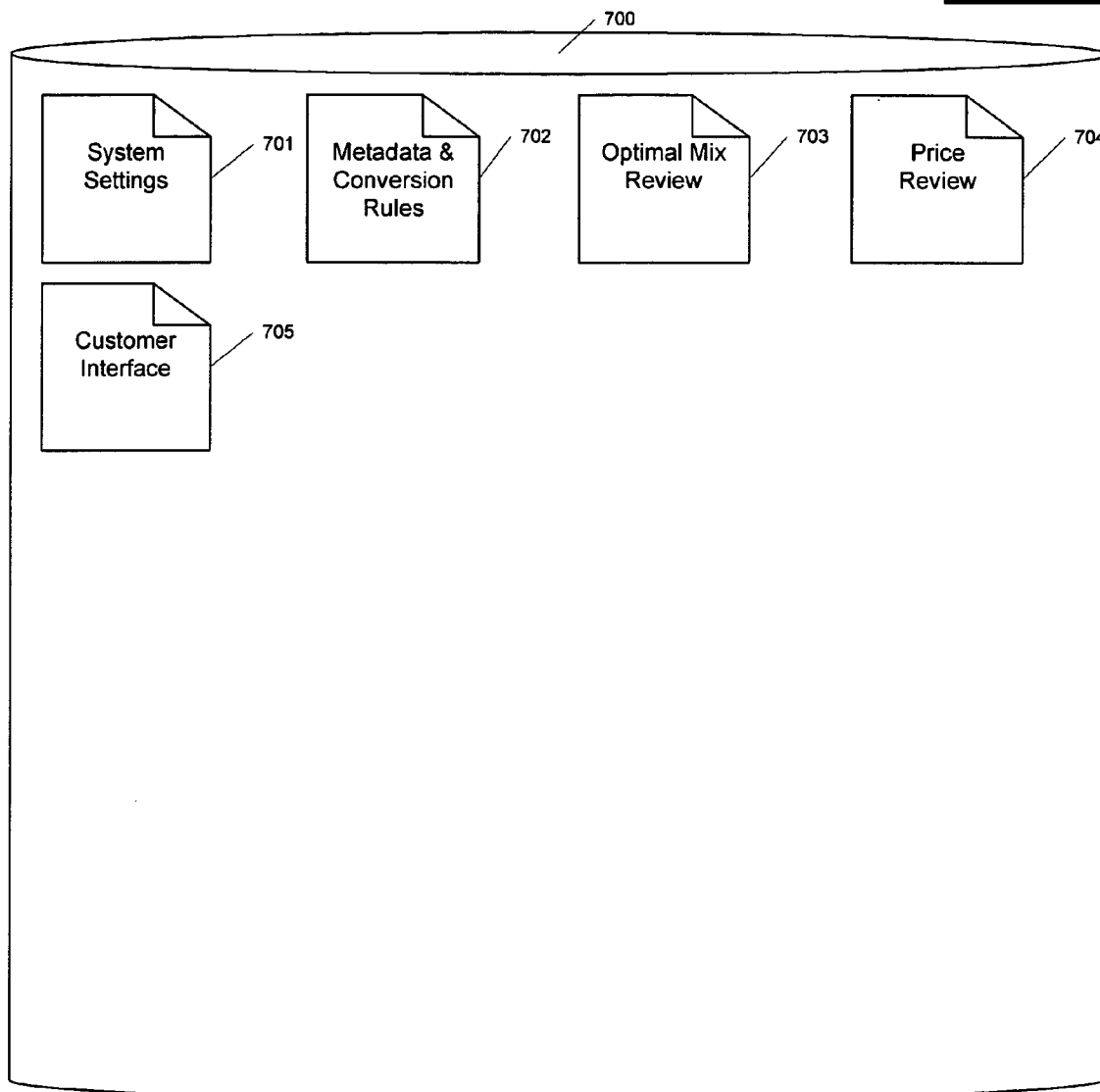


FIG. 5A

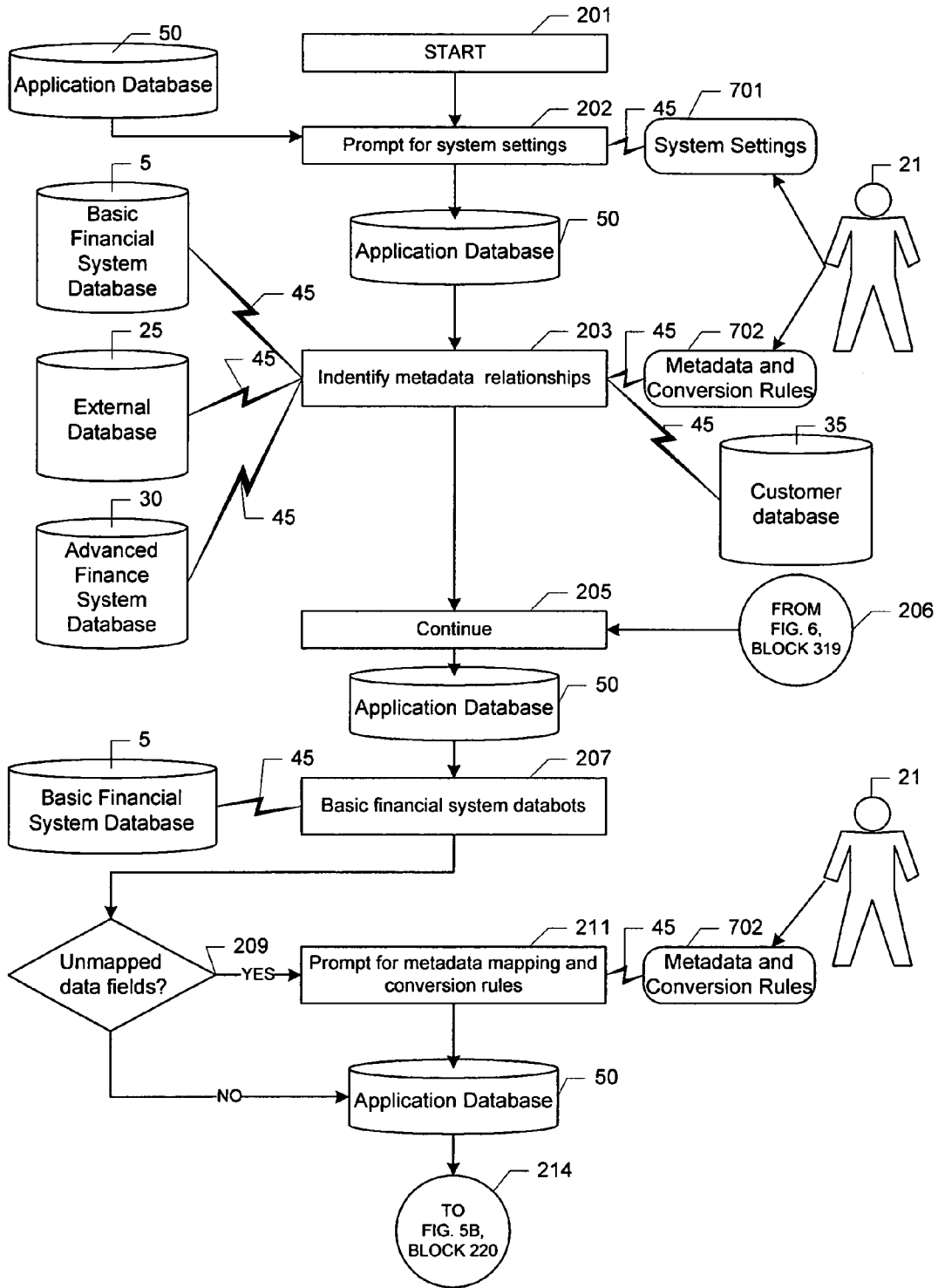


FIG. 5B

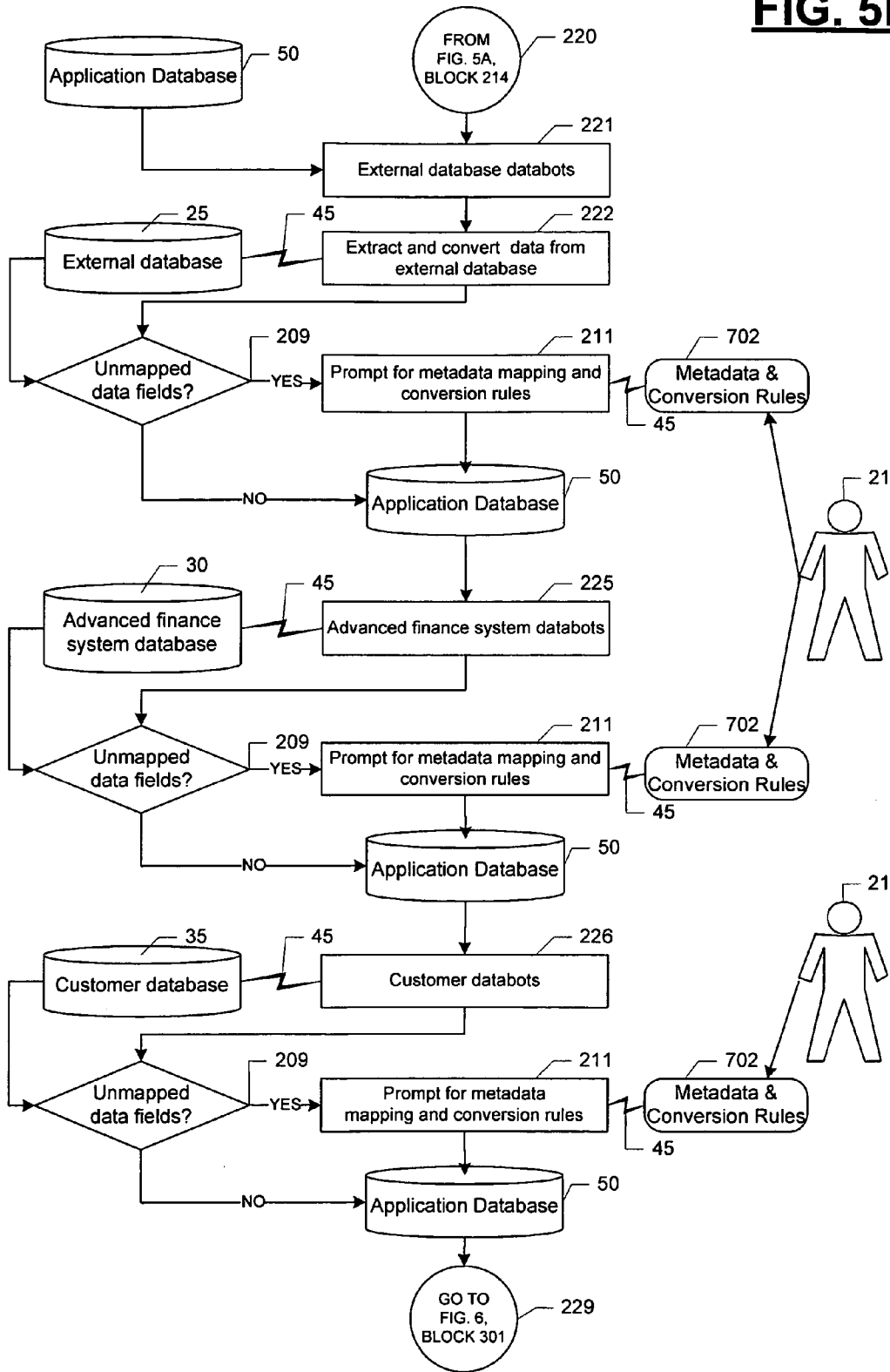
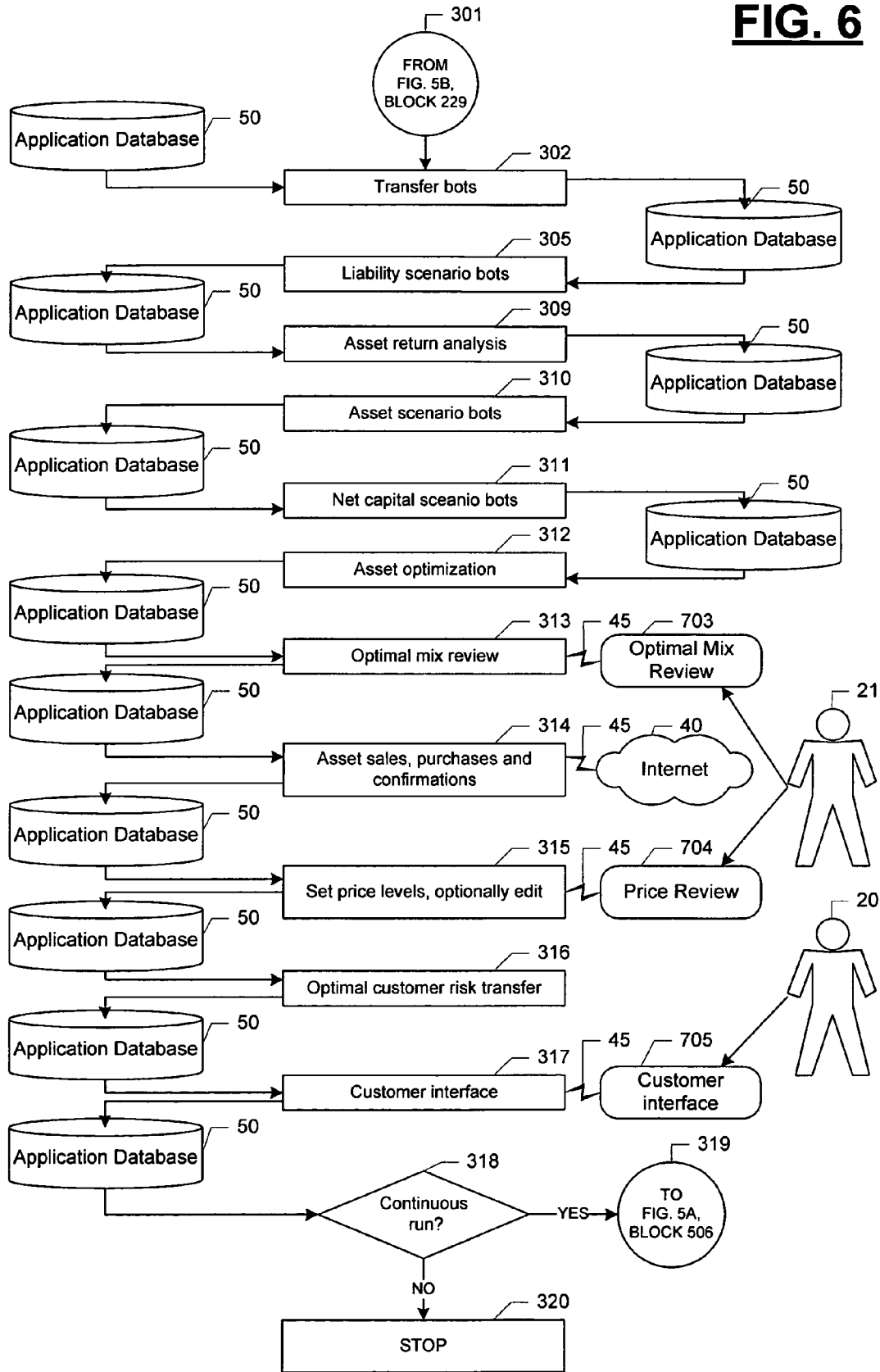


FIG. 6



ENTERPRISE RISK MANAGEMENT SYSTEM

**CONTINUATION, CONTINUATION IN PART
AND CROSS REFERENCE TO RELATED
APPLICATIONS**

[0001] This application is a continuation of application Ser. No. 10/329,172 filed Dec. 23, 2002 the disclosure of which is incorporated herein by reference and a continuation in part of application Ser. No. 09/688,983 filed Oct. 17, 2000 the disclosure of which is incorporated herein by reference. The subject matter of this application is also related to the subject matter of U.S. patent application Ser. No. 09/994,740 filed Nov. 28, 2001, U.S. patent application Ser. No. 10/012,374 filed Dec. 12, 2001, U.S. patent application Ser. No. 10/036,522 filed Jan. 7, 2002, U.S. patent application Ser. No. 10/747,471 filed Dec. 29, 2003, U.S. patent application Ser. No. 10/821,504 filed Apr. 9, 2004 and U.S. patent application Ser. No. 11/142,785 filed May 31, 2005 the disclosures of which are incorporated herein by reference.

BACKGROUND OF THE INVENTION

[0002] This invention relates to the identification, measurement and management of the risk associated with a commercial enterprise and the automated development and delivery of products and programs that help transfer said risk.

[0003] Insurance and most financial services are generally delivered in a manner that is very cumbersome. A system that would enable financial service firms to provide financial “products” like insurance, derivatives, foreign exchange, capital and credit tailored to the specific situation on a “just-in-time” would clearly be beneficial.

[0004] One of the biggest problems with achieving this goal has been that there has been no agreed upon method for analyzing risk, liquidity and foreign exchange requirements and for communicating that information to financial service firms. It is worth noting at this point that while XML is widely touted as a panacea for inter-firm communication it is only useful in establishing the language for the communication—not the substance of what is being communicated. To satisfy all the potential providers of financial services, the substance of the communication regarding risk, liquidity and foreign exchange requirements would have to overcome the limitations of traditional systems and xml.

[0005] In light of the preceding discussion, it is clear that it would be desirable to have an automated, real time system that could identify the full spectrum of risk transfer (and liquidity) needs for an enterprise in a way that that would allow financial service firms to provide “just in time” and/or real time financial products and services in a manner that is customized to the exact needs of the customers using the system.

SUMMARY OF THE INVENTION

[0006] It is a general object described herein to provide a novel and useful system for the identification, measurement and management of the risk associated with a commercial enterprise and the automated development and delivery of products and programs that help transfer said risk.

[0007] The information regarding the risk transfer needs for each customer using the system is continuously devel-

oped and communicated to the operator of the system described herein (such as an insurance company or bank) that analyzes the information for each customer in order to:

[0008] 1) Arrange swaps of risk, between customers with complementary, offsetting needs (for a fee);

[0009] 2) Arrange for risk transfers for a larger fee as required to meet the needs of each customer using the system and the profit goals (and reserve requirements) of the firm operating the system; and

[0010] 3) Complete the swaps and risk transfers that have been arranged in accordance with customer instructions.

[0011] To provide an integrated system for transferring risk, the system described herein goes on to analyze the information provided by each enterprise and the financial status of firm operating the system (hereinafter, the system operator) to determine if standby credit lines and/or re-insurance are required. If either of these “back-up” (aka contingent) facilities for capital are required, then the appropriate amount of standby credit and/or reinsurance is determined by the system described herein.

[0012] By eliminating many of the gaps in information available to personnel in the enterprise and the system, the system described herein enables the just-in-time provision of financial service products and services such as risk transfer that are tailored to the exact needs of the enterprise. The electronic linkage also eliminates the time lag that prevents many from companies from obtaining the risk reduction products they need

BRIEF DESCRIPTION OF DRAWINGS

[0013] These and other objects, features and advantages described herein will be more readily apparent from the following description of the preferred embodiment of the invention in which:

[0014] **FIG. 1** is a block diagram showing the major processing steps described herein;

[0015] **FIG. 2** is a diagram showing the files or tables in the application database (50) described herein that are utilized for data storage and retrieval during the processing in the innovative risk transfer system;

[0016] **FIG. 3** is a block diagram of an implementation described herein;

[0017] **FIG. 4** is a diagram showing the data windows that are used for receiving information from and transmitting information to the customer (20) during system processing;

[0018] **FIG. 5**, is block diagrams showing the sequence of steps in the present invention used for specifying system settings and for initializing and operating the data bots that extract, aggregate, store and manipulate information utilized in system processing from: the basic financial system, advanced financial system, customers and external databases; and

[0019] **FIG. 6** is a block diagram showing the sequence in steps in the present invention used in the collaborative, on-line development and delivery of customized risk transfer programs.

**DETAILED DESCRIPTION OF THE
PREFERRED EMBODIMENT**

[0020] **FIG. 1** provides an overview of the processing completed by the system for the collaborative, on-line

development and delivery of customized risk transfer programs. In accordance with the present invention, an automated method of and system (100) for collaborative, on-line development and delivery of customized risk transfer programs is provided. Processing starts in this system (100) with the specification of system settings and the initialization and activation of software data "bots" (200) that extract, aggregate, manipulate and store the internal data, external data, customer (20) input and a customer financial model required for completing system processing. The data from external databases is used to analyze generic event risks and prices on investments for the asset classes and contingent liabilities specified by the system operator (21). In the preferred embodiment, the customer financial model is created using the system described in the parent application Ser. No. 09/688,983 as required to identify the impact of the different elements of value, external factors and risks on customer financial performance and value. However, any other method or system for developing this data could be used to the same effect. All required data is extracted via a network (45) from a basic financial system database (5), an external database (25), an advanced finance system database (30) and a customer database (35). These information extractions and aggregations may be influenced by a system operator (21) through interaction with a user-interface portion of the application software (700) that mediates the display, transmission and receipt of all information to and from browser software (800) such as the Microsoft Internet Explorer or Netscape Navigator in an access device (90) such as a phone or personal computer that the customer (20) or system operator interact with. While only one basic financial system database (5), external database (25), advanced finance system database (30) and customer database (35) is shown in FIG. 1, it is to be understood that the system (100) can extract data from an unlimited number of databases and customers via the network (45). It also to be understood that the customer (20) and the system operator (21) can operate separate access devices (90). It should also be understood that it is possible to complete a bulk extraction of data from each database (5, 25, 30 and 35) via the network (45) using data extraction applications before initializing the data bots. The data extracted in bulk could be stored in a single datamart or data warehouse where the data bots could operate on the aggregated data.

[0021] All extracted information is stored in a file or table (hereinafter, table) within an application database (50) as shown in FIG. 2. The application database (50) contains tables for storing input, extracted information and system calculations including an xml profile table (140), a bot date table (141), a customer table (142), a risk products table (143), a swaps table (144), a customer profile table (145), an exchange payout history table (146), an generic risk table (147), a liability scenario table (148), an asset position table (149), an external database table (150), an asset forecasts table (151), an asset correlation table (152), an scenario table (153), an exchange simulation table (154), a contingent capital table (155), an optimal exchange mix table (156) and an exchange premium history table (157) a system settings table (158), a metadata mapping table (159), a conversion rules table (160), a basic financial system table (161) and an advanced finance system table (162). Other combinations of tables and files can be used to the same effect. The application database (50) can optionally exist on a hard drive, a datamart, data warehouse or departmental warehouse. The

system described herein has the ability to accept and store supplemental or primary data directly from user input, a data warehouse or other electronic files in addition to receiving data from the customer databases described previously.

[0022] As shown in FIG. 3, the preferred embodiment described herein is a computer system (100) illustratively comprised of a user-interface personal computer (110) connected to an application-server personal computer (120) via a network (45). The application server personal computer (120) is in turn connected via the network (45) to a database-server personal computer (130). The user interface personal computer (110) is also connected via the network (45) to an internet browser appliance (90) that contains browser software (800) such as Microsoft Internet Explorer or Netscape Navigator.

[0023] The database-server personal computer (130) has a read/write random access memory (131), a hard drive (132) for storage of the application database (50), a keyboard (133), a communications bus card containing all required adapters and bridges (134), a display (135), a mouse (136) and a CPU (137).

[0024] The application-server personal computer (120) has a read/write random access memory (121), a hard drive (122) for storage of the non-user-interface portion of the enterprise portion of the application software (200 and 300) described herein, a keyboard (123), a communications bus containing all required adapters and bridges (124), a display (125), a mouse (126), a CPU (127) and a printer (128). While only one client personal computer is shown in FIG. 3, it is to be understood that the application-server personal computer (120) can be networked to fifty or more client personal computers (110) via the network (45). The application-server personal computer (120) can also be networked to fifty or more server, personal computers (130) via the network (45). It is to be understood that the diagram of FIG. 3 is merely illustrative of one embodiment described herein as the system (100) and application software (200, 300 and 700) could reside on a single computer or any number of computers that are linked together using a network. In a similar manner the system operator (21) and/or the customer (20) could interface directly with one or more of the computers in the system (100) instead of using an access device (90) with a browser (800) as described in the preferred embodiment.

[0025] The user-interface personal computer (110) has a read/write random access memory (111), a hard drive (112) for storage of a client data-base (49) and the user-interface portion of the application software (700), a keyboard (113), a communications bus containing all required adapters and bridges (114), a display (115), a mouse (116), a CPU (117) and a printer (118).

[0026] The application software (200, 300 and 700) controls the performance of the central processing unit (127) as it completes the calculations required to support the collaborative development and implementation of a risk transfer program. In the embodiment illustrated herein, the application software program (200, 300 and 700) is written in a combination of C++ and Visual Basic® although other languages can be used to the same effect. The application software (200, 300 and 700) can use Structured Query Language (SQL) for extracting data from the different databases (5, 25, 30 and 35). The customer (20) and system

operator (21) can optionally interact with the user-interface portion of the application software (700) using the browser software (800) in the browser appliance (90) to provide information to the application software (200, 300 and 700) for use in determining which data will be extracted and transferred to the application database (50) by the data bots.

[0027] User input is initially saved to the client database (49) before being transmitted to the communication bus card (124) and on to the hard drive (122) of the application-server computer via the network (45). Following the program instructions of the application software, the central processing unit (127) accesses the extracted data and user input by retrieving it from the hard drive (122) using the random access memory (121) as computation workspace in a manner that is well known.

[0028] The computers (110, 120 and 130) shown in FIG. 3 illustratively are IBM PCs or clones or any of the more powerful computers (such as mainframe computers) or workstations that are widely available. Typical memory configurations for client personal computers (110) used with the present invention should include at least 512 megabytes of semiconductor random access memory (111) and at least a 100 gigabyte hard drive (112). Typical memory configurations for the application-server personal computer (120) used with the present invention should include at least 2056 megabytes of semiconductor random access memory (121) and at least a 250 gigabyte hard drive (122). Typical memory configurations for the database-server personal computer (130) used with the present invention should include at least 4112 megabytes of semiconductor random access memory (131) and at least a 500 gigabyte hard drive (132).

[0029] Using the system described above, customer financial data is analyzed before a comprehensive risk management program is developed and implemented for each customer. The risk reduction program development is completed in two stages. As shown in FIG. 5 the first stage of processing (block 200 from FIG. 1) programs bots to continually extract, aggregate, manipulate and store the data from user input, external databases (25) and customer databases (30) as required. Bots are independent components of the application that have specific tasks to perform. As shown in FIG. 6 the second stage of processing (block 300 from FIG. 1) analyzes customer risk profiles, determines the optimal risk transfer program for each customer, sets prices and communicates with each customer as required to complete risk reduction program development and implementation. The processing described in this application for identifying the optimal risk transfer program for each customer can optionally be completed at the enterprise level (as shown in the parent application Ser. No. 09/688,983) before data is transmitted to the system of the present invention.

System Settings and Data Bots

[0030] The flow diagrams in FIG. 5 details the processing that is completed by the portion of the application software (200) that obtains systems settings from the system operator (21) before extracting, aggregating and storing the information required for system operation from a basic financial system database, an external database (25), and advanced finance system database (30) and a customer database (35).

[0031] System processing starts in a block 201, FIG. 5A, which immediately passes processing to a software block

202. The software in block 202 prompts the system operator (21) via the system settings data window (701) to provide system setting information. The system setting information entered by the system operator (21) is transmitted via the network (45) back to the application server (120) where it is stored in the system settings table (158) in the application database (50) in a manner that is well known. The specific inputs the system operator (21) is asked to provide at this point in processing are shown in Table 1.

TABLE 1

1. Continuous run, if yes, frequency? (hourly, daily, weekly, monthly or quarterly)
2. Account structure hierarchy
3. Metadata standard (XML, MS OIM, MDC)
4. Location of account structure
5. Location of basic financial system database and metadata
6. Location of advanced finance system database and metadata
7. Location of customer database(s) and metadata
8. Location of external database(s) and metadata
9. Base currency
10. Asset classes of interest
11. Contingent capital alternatives
12. Default missing data procedure
13. Maximum time to wait for user input
14. Confidence interval for risk reduction programs

The software in block 202 uses the current system date to determine the time periods (months) that require data to complete the development of risk transfer programs. After the date range is calculated, it is stored in the system settings table (158). In the preferred embodiment data is obtained for the three year period before and the three year forecast period after the current date. The system operator (21) also has the option of specifying the data periods that will be used for completing system calculations.

[0032] After the storage of system setting data is complete, processing advances to a software block 203. The software in block 203 prompts the system operator (21) via the metadata and conversion rules window (702) to map metadata using the standard previously specified by the system operator (21) (XML, Microsoft Open Information Model or the Metadata Coalitions specification) from the basic financial system database (5), the external database (25), the advanced financial system database (30) and the customer database (35) to the enterprise hierarchy stored in the system settings table (158) and to the pre-specified fields in the metadata mapping table (159). Pre-specified fields in the metadata mapping table include, the revenue, expense and capital components and sub-components for the exchange and pre-specified fields for expected value drivers. Because the bulk of the information being extracted is financial information, the metadata mapping often takes the form of specifying the account number ranges that correspond to the different fields in the metadata mapping table (159). Table 2 shows the base account number structure that the account numbers in the other systems must align with. For example, using the structure shown below, the revenue component for the enterprise could be specified as enterprise 01, any department number, accounts 400 to 499 (the revenue account range) with any sub-account.

TABLE 2

	Account Number			
	01 -	902 (any) -	477-	86 (any)
Segment	Enterprise	Department	Account	Sub-account
Subgroup	Workstation	Marketing	Revenue	Singapore
Position	4	3	2	1

As part of the metadata mapping process, any database fields that are not mapped to pre-specified fields are defined by the system operator (21) as component of value, elements of value or non-relevant attributes and “mapped” in the metadata mapping table (159) to the corresponding fields in each database in a manner identical to that described above for the pre-specified fields. After all fields have been mapped to the metadata mapping table (159), the software in block 203 prompts the system operator (21) via the metadata and conversion rules window (702) to provide conversion rules for each metadata field for each data source. Conversion rules will include information regarding currency conversions and conversion for units of measure that may be required to accurately and consistently analyze the data. The inputs from the system operator (21) regarding conversion rules are stored in the conversion rules table (160) in the application database (50). When conversion rules have been stored for all fields from every data source, then processing advances to a software block 204.

[0033] The software in block 204 checks the bot date table (141) and deactivates any basic financial system data bots with creation dates before the current system date and retrieves information from the system settings table (158), metadata mapping table (159), conversion rules table (160), the asset position table (149) and the basic financial system table (161). The software in block 204 then initializes data bots for each field in the metadata mapping table (159) that mapped to the basic financial system database (5) in accordance with the frequency specified by system operator (21) in the system settings table (158). Bots are independent components of the application that have specific tasks to perform. In the case of data acquisition bots, their tasks are to extract and convert data from a specified source and then store it in a specified location. Each data bot initialized by software block 204 will store its data in the asset position table (149) or the basic financial system table (161). Every data acquisition bot for every data source contains the information shown in Table 3.

TABLE 3

1. Unique ID number (based on date, hour, minute, second of creation)
2. The data source location
3. Mapping information
4. Timing of extraction
5. Conversion rules (if any)
6. Storage Location (to allow for tracking of source and destination events)
7. Creation date (date, hour, minute, second)

After the software in block 204 initializes all the bots for the basic financial system database, the bots extract and convert data in accordance with their preprogrammed instructions in accordance with the frequency specified by system operator (21) in the system settings table (158). As each bot extracts

and converts data from the basic financial system database (5), processing advances to a software block 209 before the bot completes data storage. The software in block 209 checks the basic financial system metadata to see if all data for all fields have been extracted and that there are metadata assignments for all extracted data. If the software in block 209 finds no unmapped data fields, then the extracted, converted data are stored in the asset position table (149) or the basic financial system table (161). Alternatively, if there are unmapped data fields, then processing advances to a block 211. The software in block 211 prompts the system operator (21) via the metadata and conversion rules window (702) to provide metadata and conversion rules for each new field. The information regarding the new metadata and conversion rules is stored in the metadata mapping table (159) and conversion rules table (160) while the extracted, converted data are stored in the asset position table (149) or the basic financial system table (161). It is worth noting at this point that the activation and operation of bots where all the fields have been mapped to the application database (50) continues. Only bots with unmapped fields “wait” for user input before completing data storage. The new metadata and conversion rule information will be used the next time bots are initialized in accordance with the frequency established by the system operator (21). In either event, system processing passes on to a software block 221.

[0034] The software in block 221 checks the bot date table (141) and deactivates any external database data bots with creation dates before the current system date and retrieves information from the generic risk table (147), external database table (150), system settings table (158), metadata mapping table (159) and conversion rules table (160). The software in block 221 then initializes data bots for each field in the metadata mapping table (159) that mapped to the external database (25) in accordance with the frequency specified by system operator (21) in the system settings table (158). Bots are independent components of the application that have specific tasks to perform. In the case of data acquisition bots, their tasks are to extract and convert data from a specified source and then store it in a specified location. Each data bot initialized by software block 221 will store its data in the generic risk table (147) or the external database table (150). After the software in block 221 initializes all the bots for the advanced finance system database, the bots extract and convert data in accordance with their preprogrammed instructions in accordance with the frequency specified by system operator (21) in the system settings table (158). As each bot extracts and converts data from the external database (25), processing advances to a software block 209 before the bot completes data storage. The software in block 209 checks the advanced finance system metadata to see if all data for all fields have been extracted and that there are metadata assignments for all extracted data. If the software in block 209 finds no unmapped data fields, then the extracted, converted data are stored in the generic risk table (147) or external database table (150). Alternatively, if there are unmapped data fields, then processing advances to a block 211. The software in block 211 prompts the system operator (21) via the metadata and conversion rules window (702) to provide metadata and conversion rules for each new field. The information regarding the new metadata and conversion rules is stored in the metadata mapping table (159) and conversion rules table (160) while the extracted, converted data are stored in the

generic risk table (147) or external database table (150). It is worth noting at this point that the activation and operation of bots where all the fields have been mapped to the application database (50) continues. Only bots with unmapped fields “wait” for user input before completing data storage. The new metadata and conversion rule information will be used the next time bots are initialized in accordance with the frequency established by the system operator (21). In either event, system processing passes on to a software block 225.

[0035] The software in block 225 checks the bot date table (141) and deactivates any advanced finance system data bots with creation dates before the current system date and retrieves information from the system settings table (158), metadata mapping table (159), conversion rules table (160) and advanced finance system table (162). The software in block 225 then initializes data bots for each field in the metadata mapping table (159) that mapped to the advanced finance system database (30) in accordance with the frequency specified by system operator (21) in the system settings table (158). Bots are independent components of the application that have specific tasks to perform. In the case of data acquisition bots, their tasks are to extract and convert data from a specified source and then store it in a specified location. Each data bot initialized by software block 225 will store its data in the asset position table (149) or the advanced finance system table (162). After the software in block 225 initializes all the bots for the advanced finance system database, the bots extract and convert data in accordance with their preprogrammed instructions in accordance with the frequency specified by system operator (21) in the system settings table (158). As each bot extracts and converts data from the basic financial system database (5), processing advances to a software block 209 before the bot completes data storage. The software in block 209 checks the advanced finance system metadata to see if all data for all fields have been extracted and that there are metadata assignments for all extracted data. If the software in block 209 finds no unmapped data fields, then the extracted, converted data are stored in the asset position table (149) or the advanced finance system table (162). Alternatively, if there are unmapped data fields, then processing advances to a block 211. The software in block 211 prompts the system operator (21) via the metadata and conversion rules window (702) to provide metadata and conversion rules for each new field. The information regarding the new metadata and conversion rules is stored in the metadata mapping table (159) and conversion rules table (160) while the extracted, converted data are stored in asset position table (149) or the advanced finance system table (162). It is worth noting at this point that the activation and operation of bots where all the fields have been mapped to the application database (50) continues. Only bots with unmapped fields “wait” for user input before completing data storage. The new metadata and conversion rule information will be used the next time bots are initialized in accordance with the frequency established by the system operator (21). In either event, system processing passes on to a software block 226.

[0036] The software in block 226 checks the bot date table (141) and deactivates any customer database data bots with creation dates before the current system date and retrieves information from the system settings table (158), metadata mapping table (159), conversion rules table (160) and customer table (142). The software in block 226 then initializes

data bots for each field in the metadata mapping table (159) that mapped to the customer database (35) in accordance with the frequency specified by system operator (21) in the system settings table (158). Bots are independent components of the application that have specific tasks to perform. In the case of data acquisition bots, their tasks are to extract and convert data from a specified source and then store it in a specified location. Each data bot initialized by software block 226 will extract the model of customer financial performance by element of value, factor and risk and the confidence interval for risk reduction programs specified by the customer. The bot will then store this data in the customer profile table (145). After the software in block 226 initializes all the bots for the advanced finance system database, the bots extract and convert data in accordance with their preprogrammed instructions in accordance with the frequency specified by system operator (21) in the system settings table (158). As each bot extracts and converts data from the customer database (25), processing advances to a software block 209 before the bot completes data storage. The software in block 209 checks the advanced finance system metadata to see if all data for all fields have been extracted and that there are metadata assignments for all extracted data. If the software in block 209 finds no unmapped data fields, then the extracted, converted data are stored in the customer profile table (145). Alternatively, if there are unmapped data fields, then processing advances to a block 211. The software in block 211 prompts the system operator (21) via the metadata and conversion rules window (702) to provide metadata and conversion rules for each new field. The information regarding the new metadata and conversion rules is stored in the metadata mapping table (159) and conversion rules table (160) while the extracted, converted data are stored in the customer profile table (145). It is worth noting at this point that the activation and operation of bots where all the fields have been mapped to the application database (50) continues. Only bots with unmapped fields “wait” for user input before completing data storage. The new metadata and conversion rule information will be used the next time bots are initialized in accordance with the frequency established by the system operator (21). In either event, system processing passes on to software block 301.

Risk Exchange

[0037] The flow diagram in FIG. 6 details the processing that is completed by the portion of the application software (300) that analyzes information from a number of customers and arranges for risk “swaps” and/or the sale of risk transfer products to each customer at a price that meets the profit goals and reserve requirements of the company operating the risk exchange. The description below will follow the processing and activities of the system described herein when one new customer profile is transmitted to the exchange.

[0038] System processing in this portion of the application software (300) begins in a block 302. The software in block 302 checks the bot date table (141) and deactivates any transfer bots with creation dates before the current system date for the customer transmitting data to the exchange. The software in block 302 then retrieves the information from the xml profile table (140), the customer table (142), the risk products table (143), the swaps table (144) and the customer

profile table (145) as required to initialize transfer bots for the customer transmitting a summary profile to the exchange.

[0039] Bots are independent components of the application that have specific tasks to perform. In the case of transfer bots, their primary tasks are to identify swaps, existing product and new products that can be used to satisfy the risk transfer needs of the customer transmitting data to the exchange. For example, if one customer has a significant risk from oil prices dropping (a heating oil company, for example) and another customer faces a significant risk when oil prices rise (a trucking company, for example), then the transfer bot will identify the offsetting risk factors and record a swap. If the risk transfer can be completed by both an existing risk transfer product and a swap, then preference is given to the swap. Every transfer bot contains the information shown in Table 4.

TABLE 4

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Risk factor
6. Type: Swap, existing product or (potential) new product
7. Amount(s)
8. Date(s)
9. Customer 1 (for swaps only)
... to
9 + n. Customer n (for swaps only)

After the transfer bot identifies the swaps, existing products and new products that will satisfy the needs of the enterprise for risk transfer the results are saved to the application database (50). Information on swaps is saved on the swaps table (144) and the customer profile table (145) and information on new products is saved in the risk products table (143) without a price. The price for new products will be established later in the processing. After data storage is complete, processing advances to a software block 305.

[0040] The software in block 305 checks the bot date table (141) and deactivates any liability scenario bots with creation dates before the current system date. The software in block 305 then retrieves the information from the xml profile table (140), the customer table (142), the risk products table (143), the swaps table (144), the customer profile table (145), the exchange payout history table (146), the generic risk table (147) and the exchange premium history table (156) as required to initialize new liability scenario bots. Bots are independent components of the application that have specific tasks to perform. In the case of liability scenario bots, their primary tasks are to create a series of scenarios estimating the net payout (premiums minus payout=net payout) associated the risks that may be transferred via swaps or insurance from all customers. There are two types of scenarios developed at this stage of processing, normal scenarios and extreme scenarios. The scenarios are developed by combining the information and statistics from summary profiles transmitted by the customers of the exchange with the exchange payout history, the exchange premium history and generic risk information obtained from the external database (25). Every liability scenario bot

activated in this block contains the information shown in Table 5.

TABLE 5

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Type: extreme or normal

After the liability scenario bots are initialized, they retrieve the required information from the xml profile table (140), the customer table (142), the risk products table (143), the swaps table (144), the customer profile table (145), the exchange payout history table (146), the generic risk table (147), the external database table (150), the basic financial system table (161), the advanced finance system table (162) and the exchange premium history (156) before generating a series of net payout scenarios that are appropriate for the type of analysis being completed—extreme or normal. The bot saves the scenarios in the liability scenario table (148) in the application database (50) and processing advances to a block 309.

[0041] The software in block 309 continually completes analyses similar to those completed by the analysis bots in the enterprise portion of the parent application Ser. No. 09/688,983. The software in this block uses the publicly available information stored in the external database table (150) to complete the analyses shown in Table 6 for each equity investment listed in the asset position table (149) and described in data obtained from the external database (25).

TABLE 6

1. Identify market value factors causing changes in the equity market price;
2. Forecast the value of the current operation for the company as a function of the causal factors identified in 1 and prior performance using forecasting method for revenue, expense and capital change similar to that described in related U.S. Pat. No. 5,615,109;
3. Forecast the allocation of industry real options to the company on the basis of relative causal intangible element of value strength using forecasting method similar to that described in related U.S. Pat. No. 5,615,109; and
4. Forecast the income (dividends) provided by the equity as a function of the causal factors identified in 1 and prior performance

The results of the first three forecasts (items 2, 3 and 4 from Table 6) are saved in the asset forecasts table (151) in the application database (50) and the market value factors (item 1 from Table 6) are saved with the appropriate equity in the asset position table (149). The software in this block uses the publicly available information stored in the external database table (150) to complete the analyses shown in Table 6 for each income generating investments (i.e. bonds or real estate) listed in the asset position table (149) and described in data obtained from the external database (25). The software in block 309 then analyzes the covariance between the causal factors for each of the assets to determine the covariance between these assets under both normal and extreme conditions. The results of these analyses are then stored in the asset correlation table (152) before processing advances to a block 310.

[0042] The software in block 310 checks the bot date table (141) and deactivates any scenario bots with creation dates before the current system date. The software in block 310 then retrieves the information from the asset position table (149), the external database table (150) and the asset correlation table (152) as required to initialize the scenario bots. Bots are independent components of the application that have specific tasks to perform. In the case of scenario bots, their primary task is to identify likely scenarios for the evolution of the causal market value factors. The scenario bots use information from the external databases to obtain forecasts for individual causal factors before using the covariance information stored in the asset correlation table (152) to develop scenarios for the other causal factors under normal and extreme conditions. Every scenario bot activated in this block contains the information shown in Table 7.

TABLE 7

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Type: Normal or extreme

After the scenario bots are initialized, they retrieve the required information and develop a variety of normal and extreme scenarios as described previously. After the scenario bots complete their calculations they save the resulting scenarios in the scenario table (153) in the application database (50) and processing advances to a block 311.

[0043] The software in block 311 checks the bot date table (141) and deactivates any net capital scenario bots with creation dates before the current system date. The software in block 311 then retrieves the information from the liability scenario table (148), and the scenario table (153) as required to initialize net capital scenarios bots. Bots are independent components of the application that have specific tasks to perform. In the case of net capital scenario bots, their primary task is to run four different types of simulations for the exchange. The net capital scenario bots run Monte Carlo simulations of the exchange financial performance using the two types of scenarios generated by the asset and liability scenario bots—normal and extreme. The net capital scenario bots also run an unconstrained genetic algorithm simulation that evolves to the most negative scenario and simulations specified by regulatory agencies. Every net capital scenario bot activated in this block contains the information shown in Table 8.

TABLE 8

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Type: Normal, extreme, genetic algorithm or compliance

After the net capital scenario bots are initialized, they retrieve the required information and simulate the financial performance of the risk exchange under the different scenarios.

[0044] After the net capital scenarios complete their calculations, the resulting forecasts are saved in the exchange

simulation table (154) in the application database (50) and processing advances to a block 312.

[0045] The software in block 312 checks the bot date table (141) and deactivates any asset optimization bots with creation dates before the current system date. The software in block 312 then retrieves the information from the asset position table (149), the external database table (150), the asset forecasts table (151), the asset correlation table (152), the scenario table (153), the exchange simulation table (154) and the advanced finance systems table (162) as required to initialize asset optimization bots. Bots are independent components of the application that have specific tasks to perform. In the case of asset optimization bots, their primary task is to determine the optimal mix of assets and contingent capital purchases (purchase reinsurance and/or other contingent capital purchases, etc.) for the exchange under each scenario using a linear programming optimization algorithm that is constrained by any limitations imposed by regulatory requirements. A multi-criteria optimization is also run at this stage to determine the best mix for maximizing value under combined normal and extreme scenarios. A penalty function for asset liability mismatch can be added as required to minimize the difference between asset and liability lives. Other optimization algorithms can be used at this point to achieve the same result. Every asset optimization bot activated in this block contains the information shown in Table 9.

TABLE 9

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Type: Normal, extreme or combined

After the asset optimization bots complete their analyses, the resulting asset and contingent capital mix for each set of scenarios and the combined analysis is saved in the optimal exchange mix table (156) in the application database (50) and the revised simulations are saved in the exchange simulation table (154) before processing passes to a software block 313.

[0046] The software in block 313 prepares and displays the optimal mix of asset purchases, asset sales and contingent capital purchases for the normal, extreme and combined scenario analysis using the optimal mix review window (703). The optimal mix for the normal and extreme scenarios are determined by calculating the weighted average sum of the different scenarios where the weighting is determined by the relative likelihood of the scenario. The display identifies the optimal mix from the combined analysis as the recommended solution for exchange value maximization. At this point, the system operator (21) is given the option of:

- [0047] 1) Editing (adding or deleting products and activities) from the recommended solution;
- [0048] 2) Selecting the optimal mix from the normal scenarios;
- [0049] 3) Selecting and then editing the optimal mix from the normal scenarios;
- [0050] 4) Selecting the optimal mix from the extreme scenarios;

[0051] 5) Selecting and then editing the optimal mix from the extreme scenarios; or

[0052] 6) Leaving the default choice in place.

[0053] After the system operator (21) has finished the review and the optional edit of the selected mix, any changes are saved in the optimal exchange mix table (156) in the application database (50) before processing advances to a software block 314.

[0054] The software in block 314 compares the new optimal mix to the existing asset position stored in the asset position table (149) and orders are generated to purchase assets, sell assets and/or purchase contingent capital as required to bring the current asset position in line with the new optimal mix. These orders are then transmitted via a network (45) to other institutions and exchanges on the Internet (40). When the order confirmations are received, the asset position table (149) is updated with the new information and processing advances to a block 315. It is worth noting at this point that the processing described for the previous blocks in this section (302, 305, 109, 310, 311, 312, 313 and 314) could also be used to manage an investment portfolio on a stand alone basis.

[0055] The software in block 315 prepares and displays the proposed prices for the risk transfer products and the swaps that are going to be offered to the customer using the price review window (704). The list prices from the risk products table (143) are used for the existing risk products. Pricing for swaps are calculated by marking up the cost of the swap by a standard percentage. The software in block 315 marks up the calculated breakeven price for any new risk transfer products that were proposed by the bots in block 302. At this point, the system operator (21) is given the option of:

[0056] 1) Editing the recommended prices for any and all of the risk transfers—swaps, existing products and new products;

[0057] 2) Accepting the recommended prices; or

[0058] 3) Removing some of swaps and/or risk transfer products from the list.

[0059] After the system operator (21) completes the review, all price changes and the prices for any new risk transfer products are saved in the risk products table (143) before processing advances to a block 316.

[0060] The software in block 316 continually runs an analysis to define the optimal risk reduction strategy for the normal and extreme scenarios for each customer. It does this by first retrieving data from the xml profile table (140), the customer table (142), the risk products table (143), the swaps table (144), the customer profile table (145), the exchange payout history table (146), the generic risk table (147), the external database table (150) and the scenario table (153)—the information required to initialize the optimization algorithm. The software in the block uses a linear program that uses the financial model for each customer under the range of conditions expected for each scenario to determine the optimal risk transfer program (swaps, derivative purchases, insurance purchases, etc.) within the specified confidence interval (the confidence interval specified by the system operator (21) is used if the customer has not specified a confidence interval). A multi criteria optimization

determines the best mix for reducing the risk under a combined normal and extreme scenario. Other optimization algorithms and simulations can be used at this point to the same effect. The optimizations consider the effect of changes in the cost of capital on the optimal risk transfer solution. The resulting mix of product purchases and swaps for each scenario (normal and extreme) and the combined analysis is saved in the customer profile table (145) in the application database (50) before processing passes to a software block 317. The shadow prices from these optimizations are also stored in the risk products table (143) for use in identifying new risk reduction products that the system operator (21) may choose to offer at a later date. This information can also be used to modify pricing by customer.

[0061] The software in block 317 uses the customer interface window (705) to display the information regarding the optimal risk transfer program for the customer and the pricing for the products and swaps that will be used to transfer the risks identified in the optimal risk transfer program. This information could optionally be transmitted to the customer in a summary xml format that is similar to the one initially transmitted to the exchange by the customer. The customer (20) can reject, edit and/or accept the proposed mix of products and swaps that are displayed. The software in block 317 accepts and confirms orders, updates the information contained in the risk products table (143), the swaps table (144), the customer profile table (145) and the exchange premium history table (157) to reflect the accepted and confirmed orders. The software in block 317 also accepts input from the customer (20) regarding any new losses that the customer may have experienced. The software in block 317 verifies the loss is for an insured risk, updates the customer profile table (145), updates the exchange payout history table (146) and arranges for payment of the claim in a manner that is well known. This processing is continues until the customer (20) indicates that the session is complete. System processing advances to a software block 318.

[0062] The software in block 318 checks the system settings table (158) to determine if the system (100) is operating in continuous mode. If the system is operating in a continuous mode, then processing returns to block 205 and the processing described above is repeated. Alternatively, if the system is not operating in continuous mode, then processing advances to a software block 320 and stops.

[0063] Thus, the reader will see that the system and method described above transforms extracted transaction data, corporate information, information from external databases and information from the internet into detailed risk analyses and risk transfer programs specifically tailored to each customer using the system. The level of detail, breadth and speed of the risk analysis allows customers and managers of the system to manage their risks in a fashion that is superior to the method currently available to users of existing risk analysis systems and traditional insurance products.

[0064] Because the profiles used in the system (100) provide a comprehensive picture of the financial status of the companies transferring risk through the exchange, the system and method described herein can be used with essen-

tially no modifications to provide an on-line transfer system for:

- [0065] 1. Foreign exchange;
- [0066] 2. Capital (aka liquidity);
- [0067] 3. Any combination of foreign exchange, capital and risk.

The system described herein could be used to manage transfers of ownership rights alone or in combination with foreign exchange, liquidity and risk.

[0068] While the above description contains many specificities, these should not be construed as limitations on the scope of the invention, but rather as an exemplification of one preferred embodiment thereof. Accordingly, the scope of the invention should be determined not by the embodiment illustrated, but by the appended claims and their legal equivalents.

1. A risk method, comprising:

aggregating data from a plurality of management systems for a commercial enterprise, identifying a plurality of risks that may have an impact on a financial performance of said enterprise by analyzing at least a portion of said data, and

completing a series of tasks related to said risks where the tasks are selected from the group consisting of measuring said risks, identifying an optimal set of risk transfer transactions, identifying an optimal mode for operating one or more risk management control programs, completing an optimal set of risk transfer transactions, reporting said risks and combinations thereof

where a plurality of risks further comprise event risks and risks selected from the group consisting of contingent liabilities, strategic risks, variability risks, volatility and combinations thereof.

2. The method of claim 1 wherein identifying an optimal set of risk transfer transactions further comprise identifying a set of transactions selected from the group consisting of insurance policy transactions, swap transactions, derivative transactions, contingent capital transactions and combinations thereof.

3. The method of claim 1 wherein measuring a plurality of identified risks further comprises developing a plurality of scenarios selected from the group consisting of normal, extreme, regulatory and combinations thereof for a commercial enterprise financial performance and simulating the performance of the commercial enterprise by segment of value under each scenario as required to measure the impact of each risk from the plurality of risks where the segments of value are selected from the group consisting of current operation, real option, derivative, investment, market sentiment and combinations thereof.

4. The method of claim 1 that further comprises:

obtaining a set of measured risks for a plurality of customers, a set of regulatory requirements and a plurality of performance data for a risk transfer operation,

generating a plurality of scenarios for a risk transfer operation performance using the performance data, regulatory requirements and customer risk data;

identifying and displaying an optimal mode for risk transfer operation under each scenario, and

optionally implementing the optimal mode for a chosen scenario in an automated fashion.

5. The method of claim 4 wherein an optimal mode for risk transfer operation is a mode that maximizes risk transfer operation value while satisfying risk transfer operation regulatory requirements.

6. The method of claim 4 wherein an optimal mode for risk transfer operation is a mode that maximizes customer value while satisfying risk transfer operation regulatory requirements.

7. The method of claim 4 wherein implementing an optimal mode for the risk transfer operation includes taking actions selected from the group consisting of changing one or more contingent capital contracts, purchasing one or more contingent capital contracts, changing a duration for one or more investments, changing an investment mix, adding one or more risk transfer products, changing one or more risk transfer product specification, changing one or more risk transfer product prices, changing one or more reserve levels, completing a plurality of customer risk transfer transactions and combinations thereof.

8. A computer readable medium having sequences of instructions stored therein, which when executed cause the processor in a computer to perform a risk method, comprising:

aggregating data from a plurality of management systems for a commercial enterprise, developing a model of enterprise financial performance by segment of value using said data; and

quantifying a plurality of risks that have an impact on the financial performance of said enterprise using said model

where the segments of value are selected from the group consisting of current operation, real option, derivative, investment, market sentiment and combinations thereof.

9. The computer readable medium of claim 8 wherein the method further comprises:

completing a series of tasks related to a plurality of quantified risks for an enterprise wherein said tasks are selected from the group consisting of identifying an optimal set of risk transfer transactions for the enterprise, identifying an optimal mode for operating one or more risk management control programs for the enterprise, completing an optimal set of risk transfer transactions for the enterprise, reporting said enterprise risks and combinations thereof

10. The computer readable medium of claim 8 wherein a plurality of risks further comprise event risks and risks selected from the group consisting of contingent liabilities, strategic risks, variability risks, volatility and combinations thereof.

11. The computer readable medium of claim 8 wherein identifying an optimal set of risk transfer transactions further comprise identifying a set of transactions selected from the group consisting of insurance policy transactions, swap transactions, derivative transactions, contingent capital transactions and combinations thereof.

12. The computer readable medium of claim 8 wherein quantifying a plurality of risks further comprises developing

a plurality of scenarios selected from the group consisting of normal, extreme, regulatory and combinations thereof for a commercial enterprise financial performance and simulating the performance of the commercial enterprise by segment of value under each scenario as required to measure the impact of each risk from the plurality of risks.

13. The computer readable medium of claim 8 wherein developing a model of enterprise financial performance by segment of value further comprises developing a model of the impact of one or more elements of value and a plurality of external factors on an enterprise financial performance by segment of value where a plurality of external factors are selected from a group consisting of the ratio of actual earnings to expected earnings, a commodity price, an inflation rate, a gross national product (gnp) growth rate, a market volatility measure, volatility vs. industry average volatility, an interest rate, an interest rate change, an insider trading direction, an insider trading level, a consumer confidence level an unemployment rate and combinations thereof.

14. The computer readable medium of claim 8, wherein the method further comprises:

obtaining a set of measured risks for a plurality of customers, a set of regulatory requirements and a plurality of performance data for a risk transfer operation,

generating a plurality of scenarios for a risk transfer operation performance using the performance data, regulatory requirements and customer risk data;

identifying and displaying an optimal mode for risk transfer operation under each scenario, and

optionally implementing the optimal mode for a chosen scenario in an automated fashion.

15. A risk system, comprising:

networked computers each with a processor having circuitry to execute instructions; a storage device available to each processor with sequences of instructions stored therein, which when executed cause the processors to:

aggregate data from a plurality of management systems for a plurality of commercial enterprise customers,

develop a model of enterprise financial performance by segment of value for each customer using said data;

quantify a plurality of risks that have an impact on the financial performance of each enterprise using said model

obtain a set of regulatory requirements and a plurality of performance data for a risk transfer operation,

generate a plurality of scenarios for a risk transfer operation performance using the performance data, regulatory requirements and quantified customer risk data; and

identify and display an optimal mode for risk transfer operation under each scenario.

16. The system of claim 15 wherein an optimal mode for risk transfer operation is a mode that maximizes risk transfer operation value while satisfying risk transfer operation regulatory requirements.

17. The system of claim 15 wherein an optimal mode for risk transfer operation is a mode that maximizes customer value while satisfying risk transfer operation regulatory requirements.

18. The system of claim 15 wherein an optimal mode for risk transfer operation is implemented in an automated fashion by taking actions selected from the group consisting of changing one or more contingent capital contracts, purchasing one or more contingent capital contracts, changing a duration for one or more investments, changing an investment mix, adding one or more risk transfer products, changing one or more risk transfer product specification, changing one or more risk transfer product prices, changing one or more reserve levels, completing an optimal set of risk transfer transactions for each customer and combinations thereof.

19. The system of claim 18 wherein an optimal set of risk transfer transactions for each customer further comprise a set of transactions selected from the group consisting of insurance policy transactions, swap transactions, derivative transactions, contingent capital transactions and combinations thereof that maximize a customer value within a constraint of available capital.

20. A method for optimizing the operation of a risk transfer organization, comprising:

obtaining a plurality of data that define external market conditions, reserve requirements and financial performance for a plurality of customers for a risk transfer operation, and

identifying an optimal mode for risk transfer operation wherein the optimal mode for risk transfer operation specifies a set of actions selected from the group consisting of changing one or more contingent capital contracts, purchasing one or more contingent capital contracts, changing a duration for one or more investments, changing an investment mix, adding one or more risk transfer products, changing one or more risk transfer product specification, changing one or more risk transfer product prices, changing one or more reserve levels, completing an optimal set of risk transfer transactions for each customer and combinations thereof that will optimize risk transfer operation financial performance.

* * * * *