

ORDER-RESOURCE FULFILLMENT AND MANAGEMENT SYSTEM AND APPROACH

5 **FIELD OF THE INVENTION**

The present invention is directed to transaction processing and, more specifically, to the processing and management of interactions relating to transactions involving goods and/or services.

10 **BACKGROUND**

Transaction processing has typically involved intensive manual effort and, in instances where automatic processing has been used, intensive user intervention. For example, transaction processes involve the use of a variety of transaction documents such as orders, invoices, receipts and bills of lading (BOL). These types of transaction documents include information associated with the transaction that is used by parties to the transaction to monitor and process the transaction.

Transaction documents are electronically processed for a multitude of different types of applications. Interaction data, (e.g., electronic or physical documents) describing characteristics of a particular transaction interaction is often encountered in varied temporal order at central transaction locations that assemble these documents into logical packages for automated processing. For example, when a transaction involves the sale of a product from a seller to a buyer, there are often multiple parties to the transaction in addition to the buyer and seller, such as shippers, financial institutions, distributors and regulatory agencies (e.g., customs, taxation agencies). Each of these parties often provides one or more different types of documents that relate to the transaction. Often, the documents are not in a format that is readily discernible relative to documents from other parties, requiring extensive effort to organize the documents into categories or transactions.

A variety of transactions are particularly susceptible to processing difficulties such as those relating to document identification and categorization. For example, pre-payment
30 reconciliation and auditing for a particular transaction are often automatically carried out electronically at a central processor. Documents used for these functions can arrive at the central processor in either an untimely manner or a format that is unsuitable for identification and categorization of the documents into particular transactions. As another

example, a customs clearance document referencing a particular invoice could arrive at a central processor before the invoice, which could also arrive before the actual order arrives. Without an invoice to tie the customs clearance document, the central processor has difficulty managing the document. Similarly, without an order to attribute an invoice to, the central processor may be unable to process the invoice (*e.g.*, is unable to audit the invoice against the order).

Another type of incompatibility that has made transaction processing difficult is related to the common scenario wherein reference numbers used by different parties to identify a particular transaction are not compatible. For example, in transactions involving buyers and sellers, sellers maintain transaction data organized by reference numbers generated by the seller. Buyers typically must access the data using a seller's reference number rather than the buyer's reference number. In addition, buyers and sellers typically use different reference numbers for different characteristics of the transaction, making the monitoring and management of the transaction difficult.

As more and more documents are required to fully articulate interactions, this problem of managing documents and other interaction data and, in particular, of correlating documents and other interaction data with a proper transaction, becomes increasingly challenging. Manual parsing and categorization of these documents is expensive, time consuming and susceptible to error. Previously available automated approaches are generally limited in applicability to certain types of documents or certain inflexible methods of document identification and categorization.

Payment and billing related aspects of traditional transactions are particularly susceptible to billing errors and fraud. For example, there often is little to no connection between the delivery of goods and the billing for the delivery and/or the goods. This may result in double billing, no billing at all, or overbilling. Auditing errors that cause incorrect billing or payment may also occur. In addition, payment can often be delayed while aspects of a particular transaction are being audited and/or disputed, particularly when different transaction documents must be manually parsed and processed. For example, documents from different parties to a transaction must often be parsed and compared to relate data from one document to another in a manner that will facilitate billing. Delay associated with billing reduces working capital resources for parties to the transaction waiting for payment.

Additional costs arise as a result of existing inefficiencies in a variety of transaction-processing approaches. Many of the costs are individually small, but very large in the aggregate. For example, typical parties to transactions incur administrative costs including those relating to the costs for creating and delivering transaction documents, resolving
5 billing disputes, providing a signed copy of documents to other parties and posting accounts receivable. In addition, the cost of parsing, recognizing and categorizing documents related to these and other items add to the administrative costs of transactions.

An additional challenge to transaction management involves the inability to obtain immediate information regarding a transaction. Transaction data from one party is typically
10 not readily available to other parties involved with the transaction without direct access to private-party systems. Since the process is largely conducted manually, it is very difficult to track a transaction and real-time data is particularly difficult to come by. For example, there are various manual steps involved in order to learn of the status of shipment or payment. If a shipper wants to know if a carrier delivered the goods for a particular
15 transaction and if the payment has been made, the shipper often must contact the carrier and/or the appropriate financial institution.

The above and other difficulties in the management and coordination of transactions have presented challenges to the effective and efficient management of transactions.

20

SUMMARY

The present invention is directed to overcoming the above-mentioned challenges and others related to the types of approaches and implementations discussed above and in other applications. The present invention is exemplified in a number of implementations and applications, some of which are summarized below.

25

According to an example embodiment of the present invention, interactions are managed using an approach generally involving the use of transaction status criteria for processing payment-related aspects of the interactions.

30

In a more particular example embodiment of the present invention, interaction status conditions related to order fulfillment and related payments are monitored and/or processed as received. When changes to order fulfillment and/or related payment(s) occur and/or when new order fulfillment and payment information is received, these changes are used to

automatically update status conditions. The updated status conditions are implemented for one or more of a variety of transaction processing and management approaches.

In another example embodiment of the present invention, a transaction-processing system is adapted for managing payment resources for transactions involving merchant offerings among parties including buyers and sellers. The system includes a data storage arrangement, including one or more data storage devices at one or more locations, adapted to store order-related information for a transaction between a buyer and a seller. Often, the order-related information includes data in fields associated with historical fulfillment and billed quantities for ordered items of the transaction. A computer (*e.g.*, processing) arrangement is adapted, for a particular ordered item, to compare billed quantity information on an incoming invoice with historical fulfillment and billed quantity information for the ordered item in the data storage arrangement. If the billed quantity on the incoming invoice is no more than the difference between the historical fulfillment quantity and billed quantity information for the ordered item in the data storage arrangement, payment of the invoice is authorized. In connection with the authorization, historical billed quantities for the ordered item are updated in the data storage arrangement to include the billed quantity in the incoming invoice.

The above summary of the present invention is not intended to describe each illustrated embodiment or every implementation of the present invention. The figures and detailed description that follow more particularly exemplify these embodiments.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention may be more completely understood in consideration of the detailed description of various embodiments of the invention in connection with the accompanying drawings, in which:

FIG. 1 shows a transaction processing arrangement, according to an example embodiment of the present invention;

FIG. 2A is a flow diagram for a transaction monitoring approach, according to another example embodiment of the present invention;

FIG. 2B is another flow diagram for a transaction monitoring approach, according to another example embodiment of the present invention;

FIG. 3 is another flow diagram, related to the flow diagram shown in FIG. 2A, for an approach to categorizing data as a function of an anchor and monitoring data stored in anchor locations with automatic updating of transaction processing data, according to another example embodiment of the present invention; and

5 FIG. 4 shows an arrangement for processing claims for a plurality of transactions, according to another example embodiment of the present invention.

While the invention is amenable to various modifications and alternative forms, specifics thereof have been shown by way of example in the drawings and will be described in detail. It should be understood, however, that the intention is not necessarily to limit the
10 invention to the particular embodiments described. On the contrary, the intention is to cover all modifications, equivalents, and alternatives falling within the spirit and scope of the invention as defined by the appended claims.

DETAILED DESCRIPTION

15 The present invention is believed to be applicable to a variety of different types of transaction approaches and interactions, and has been found to be particularly useful for applications involving the processing of transactions and related aspects thereof. While the present invention is not necessarily limited to such approaches, various aspects of the invention may be appreciated through a discussion of various examples using these and
20 other contexts.

According to an example embodiment of the present invention, transaction interactions are managed using an approach that facilitates the monitoring of interaction status conditions related to order fulfillment and related payments. The management involves a variety of characteristics and can be tailored for specific implementations. For
25 example, where parties to a transaction agree upon certain business rules, the transaction interactions are managed as a function of the business rules. Business rules relating to delivery, pricing, payment and other characteristics are readily implemented in this manner for automatically managing one or more of a variety of aspects of transaction interactions. When changes to order fulfillment and/or related payment(s) occur, these changes are used
30 to automatically update status conditions.

In one implementation, resources available for processing invoices for a particular transaction (*e.g.*, express or implied order or contract) are monitored. When unbilled

resources are available, these resources are identified as those to which payment can be made. For instance, when an invoice is received for a particular quantity of goods and/or services, it is useful to compare the invoiced quantity with the quantity received to determine whether payment should be made. This approach can be used, for example, to ensure receipt of goods/services before payment is made and/or for ensuring that a sufficient quantity has been received that warrants payment (*e.g.*, where payment is effected only when a threshold quantity has been delivered or invoiced). In addition, after an invoice for an order is created, it is useful to compare the invoice against subsequent invoices to ensure that payment has not already been made for items listed. For instance, a comparison can be made to determine whether the sum of quantities billed on previous invoices is less than the fulfilled quantities (and wherein payment is subsequently made if the sum of previously billed quantities is less than fulfilled quantities). With this approach, the fulfillment status of an order is available for processing documents such as invoices, facilitating the ability to determine, for example, characteristics relating to the quantity of goods or payment therefor.

In some instances, other conditions relative to the delivery of goods and/or services are used, in addition to historical fulfillment and delivery data, for facilitating payment for transactions. For example, where transaction parties agree to payment-related terms that are tied to the time of performance (*e.g.*, goods and/or services to be delivered within a particular time period), payment for quantities fulfilled in a timely manner is authorized in accordance with agreed payment terms for on-time fulfillment. Correspondingly, payment for quantities fulfilled in an untimely manner is authorized in accordance with agreed payment terms for fulfillment that is not on time. In certain applications, the historical fulfillment data is updated as a function of the condition of goods and/or services at time of performance; where a condition of certain items is acceptable while the condition of other items is unacceptable, the historical fulfillment data is updated to reflect the amount of acceptable items. When payment is authorized as a function of historical fulfillment quantities, these fulfillment quantities reflect the condition of the items.

A variety of interactions can be monitored and managed using these approaches, with aspects of the management relating to interactions established between different parties to a transaction. Changes made to interaction data from a particular source are monitored and, when appropriate, used to update interaction data from and/or in other sources. For example, transaction-based documents applying to a common transaction can be monitored

in this manner, with an update to a document and/or data field being used to make corresponding updates to one or more other documents and/or data fields for the same common transaction, or for different transactions pertaining to a particular party to the transactions. One such example involves the monitoring of status fields relating to orders; when a particular field such as that reflecting a quantity received is updated a corresponding update can be made to related fields. This approach is applicable, for example, to related fields such as amount payable, extended price, service charge, tax and total fields.

In one implementation, related data fields and/or documents as discussed above are automatically linked for a particular transaction, in a manner commensurate with a transaction party to which the related data fields apply. In one instance, where a transaction between a buyer and a seller specifies a delivered quantity of a particular good, transaction data fields stored on behalf of each of the buyer and the seller specifying the delivered are automatically linked. In this regard, when the buyer receives goods and updates its delivered quantity data field to indicate the receipt, a corresponding update is made to the seller's delivered quantity data field to reflect the receipt acknowledged by the buyer. Many such links are selectively implemented, at the discretion of parties to each particular transaction and, in some applications, as applicable to an administrator that facilitates transaction processing on behalf of buyers and sellers.

In another implementation, business rules are used to manage order fulfillment for one or more transactions. When a particular transaction group is established, for example using a particular storage location and associated ID for the location to store documents, the transaction group is monitored for data fields associated with fulfillment. When new fulfillment data is received and/or when existing fulfillment data is altered, the fulfillment data is used to automatically process a related function. For instance, when a notice (*e.g.*, receipt of order) indicating that an order has been fulfilled is received, corresponding data fields in other documents are accordingly updated. Any response dictated by the business rules, such as the execution of payment for the fulfilled order, is then executed. For general information regarding transactions and for specific information regarding transactions and the use of a storage (*e.g.*, anchor) location to which one or more example embodiments discussed herein may apply, reference may be made to U.S. Patent Application Serial No. 10/864,761 (USBA.120PA) entitled "Automated Transaction Processing System and Approach," filed June 9, 2004 and fully incorporated herein by reference.

Referring now to FIG 1, a transaction processing arrangement 100 is adapted to automatically monitor and process transaction data, according to another example embodiment of the present invention. The transaction processing arrangement 100 includes a central transaction node 110 that is a central location for the communication and processing of transaction-related data. The central transaction node 110 is in communication with a database 112 where transaction-related information is stored. The database 112 is shown coupled to the central transaction node 110; however, in various implementations, some or all of the database is part of the central transaction node, is located at a remote location and/or includes a plurality of data storage circuits at different locations.

A plurality of user nodes 120, 122, 124, 126 and 128 are communicatively coupled with the central transaction node 110. The user nodes 120-128 may, for example, include one or more of a buyer, seller, distributor, shipper, carrier, customs agency, national security agency, financial institution or other type of individual, group or agency that would typically be involved in a transaction. These user nodes can interact with the central transaction node 110 for providing transaction-related information, such as orders, invoices, shipping documents, payment authorization, payment execution, customs documents, security documents and others that are stored in the database 112. This transaction-related information, including changes thereto, is monitored and processed by the central transaction node 110 using common attributes to automatically update information for a particular transaction. For example, when transaction data stored in the database 112 identifies a fulfillment status of an order for a particular transaction, the fulfillment status is processed and used for furthering the transaction to which the transaction data applies. Where this fulfillment status is applicable to more than one field and/or document (*i.e.*, an amount received field is applicable to an amount payable field), related fields are accordingly updated in response to one of the fields being changed.

In one implementation, transaction-related information is parsed and grouped by the central transaction node 110 for storing in the database 112 using common attributes to automatically group information from a particular transaction together, such as in U.S. Patent Application Serial No. 10/864,761, referenced above. For instance, transaction data can be grouped as a function of predefined anchor-type categories and related transaction-profile attributes. Selected ones of the transaction-profile attributes are used to form anchor

data that identifies the particular transaction and/or identifies information for the transaction, such as order, fulfillment or transaction participant information. These predefined anchor-type categories are implemented with the database 112 and the central transaction node 110 monitoring related categories; when a change in one category affects
5 another category, the other category is correspondingly updated. In addition, notification of such an update is optionally automatically sent to one of the user nodes 120-128 to alert users of the update (*e.g.*, to alert a user of a payment received or other transaction-type information).

In one implementation, the anchor data has an actual physical representation in the
10 database 112 at an anchor location, and thus can be used to access documents stored in the database and identified by the anchor. In another implementation, configuration data is used to enable parties to a transaction to define data elements of the documents that are to be used to automatically relate documents to one other and/or for automatically updating other documents. Using these approaches, the anchor data can effectively be used to
15 broadcast criteria that will allow individual transactions to latch on to a particular interaction represented by a given anchor ID. In addition, such anchor data can be used to identify related documents for updating in response to changes in transaction data from other documents having the same anchor ID. These changes are applicable to fields indicating the status of resources such as billed and/or unbilled quantities for a particular
20 order as relative to a multitude of invoices for that order.

In another implementation, the central transaction node 110 is further adapted to grant and control information exchange with the database 112 as a function of inputs received from the nodes 120-128, such as authorization inputs and document-specific inputs. When users at one of the nodes 120-128 attempt to send information to or retrieve
25 information from the central transaction node 110, authorization information from the users is used to control the information transfer. The authorization information may include, for example, access-type information (*e.g.*, a password or user ID) or simply document information that the central transaction node 110 recognizes.

When information exchange is controlled with document information and one of the
30 nodes 120-128 sends transaction data (*e.g.*, an electronic document) to the central transaction node 110, the database 112 is parsed to match the transaction data with stored information. If a match is found, the central transaction node processes the matching data to

determine whether the received transaction data belongs to the same transaction as the matching data. If so, the received transaction data is assigned an anchor ID pertaining to the matching data and is stored in the database 112 with the assigned anchor ID grouping the matching data together.

5 If a match is found but there are restrictions, the central transaction node processes the received data according to restriction rules, if any, programmed to the central transaction node. For example, where one anchor ID is assigned for each order and when there is an existing order in the database 112 under the matching anchor ID, a new anchor ID is created. The received transaction data as well as matching data that is unrestricted are
10 stored under the new anchor ID. For instance, when matching invoice data has been found but the invoice includes more than one order, a different anchor ID is created for each order, both including the same invoice but having separate orders. These and other restrictive-type approaches are programmed to the central transaction node, depending upon the implementation.

15 When a match is found, the central transaction node 110 uses business rules to identify whether the received data warrants a response, such a corresponding change to other data related to the received data or a notification to one of the user nodes 120-128. In this regard, the central transaction node 110 establishes a relationship between certain data fields in different locations. For example, by using a one-to-one correlation, data in two
20 locations can be maintained consistently where both locations identify the same or similar data, such as a quantity of ordered product, shipment status or other type of transaction characteristic. When data in one of the locations is updated, the value from that location is also updated to the other one of the locations. For example, as discussed above, certain types of data received by the central transaction node 110 can be used to correspondingly
25 update other information having a relationship with the received data. Once such approach involves the update of a resource field applicable to product or service quantities, as well as the update of corresponding payment fields relating to the quantities as discussed, for example, above in connection with the receipt of multiple invoices for a particular order.

 In another example, a relational function is used to automatically update a related
30 data field. In this example, when data in one of the locations is updated, the central transaction node 110 processes the updated data to generate an update to the other one of the locations where a one-to-one correlation does not exist. For instance, when a particular

data field applies to an order quantity and another data field applies to a monetary amount representing the order quantity, a change in order quantity is processed to generate a corresponding change in monetary amount for the order. A multitude of such relationships can be established and maintained by the central transaction node 110, some of which are
5 discussed further below.

Referring again to the matching of data received by the central transaction node with data stored in the database 112, a new anchor ID is selectively created for the received transaction data when no match is found between received transaction data and data in the database 112. This new anchor ID, along with the received transaction data, is stored in the
10 database 112 and thus establishes a new transaction. When additional transaction data matching the newly-stored data is subsequently received, the central transaction node processes the matching data as discussed above.

In another implementation, transaction data received at the central transaction node 110 is cross-referenced with security-type data stored in the database 112, prior to using the
15 received data to update corresponding fields. When transaction data matches certain security-controlled criteria, the transaction to which the transaction data belongs can be flagged or otherwise audited. For instance, where a product identification type matches a product type that is susceptible to security control, the transaction may be halted for auditing and approval by a third party. Such approval may, for example, also take the form
20 of a transaction document that is sent to the central transaction node 110, where the approval is used to further the transaction (*i.e.*, allow the corresponding fields to be updated). Security-related data received by the central transaction node 110 is monitored and used to process the transaction to which it applies, for example by updating an approval status for a particular field update to an order for which an invoice is provided.
25 Correspondingly, a field representing an approval to proceed with a transaction can be updated when approval is received.

As discussed above, the transaction processing arrangement 100 shown in FIG. 1 can be implemented in a variety of manners for processing transaction information. In one particular implementation, the transaction central transaction node 110 carries out functions
30 for addressing multilevel transactions involving more than two transaction parties. Each party interacts via the transaction processing arrangement, with fund transfers being executed with all transaction parties in a variety of manners, depending upon the situation.

In one implementation, a transaction involves a distributor or other mid-level entity that facilitates a transaction between a seller (or sellers) and buyer (or buyers). The distributor obtains transaction-related products and/or services from one or more sellers, and in turn sells these transaction-related products and/or services to end buyers. In some instances, the distributor finds buyers for products that the seller is offering, without necessarily further facilitating the transaction. In other instances, the distributor functions in one or more aspects of facilitating the transaction, such as by coordinating and/or effecting delivery, storing goods and more.

For discussion purposes with this implementation, and referring to FIG. 1, user node 120 is the seller node, user node 122 is the buyer node user node 124 is the distributor node. When the buyer 122 and the distributor 124 agree on a price for the transaction, business rules and/or other information characterizing this agreement is stored in the database 112 for use by the central transaction node 110. The distributor 124 also agrees on a price with the seller 120, with the information accordingly being stored in the database 112.

When a transaction involving goods and/or services provided by the seller 120 to the buyer 122 is facilitated by the distributor 124, the central transaction node 110 uses information stored in the database 112 to process the transaction. Effectively, the seller 120 gets paid based upon its agreement with the distributor 124, and the buyer 122 gets charged for the transaction as detailed by the buyer's agreement with the distributor 124. Any net difference between what the seller gets paid and the buyer pays is realized at the distributor. Typically, this approach involves the buyer 122 paying a higher price than what the seller is paid and the distributor receiving a markup that is the difference between the seller's price and the buyer's price. However, in some instances, this approach may also involve the distributor 124 taking a loss (*i.e.*, if the distributor has contracted with the buyer 122 for a particular price and the seller 120's price has increased). In addition, the distributor may add value to the transaction, such as via shipping, insurance, processing or other service; in this regard, the distributor may also charge the buyer directly for these value-added activities.

Payments between the buyer, seller and distributor are effected via the central transaction node 110, which interacts with financial institutions for transferring funds. For instance, the central transaction node can automatically direct funds from the buyer 122's financial institution to both the seller 120's financial institution and the distributor 124's

financial institution, based on information stored in the database 112. With this approach, the seller 120 can receive funds directly, rather than from the distributor 124, which would in turn need to collect from the buyer 122.

As discussed above, the distributor (or other mid-level entity) approach may further involve multiple buyers and/or multiple sellers. Beginning with multiple buyers, and referring again to FIG. 1 as an example, nodes 120, 122 and 124 are implemented as seller, buyer and distributor nodes as discussed above. In this instance, nodes 126 and 128 are additional buyer nodes. When all three buyer nodes 122, 126 and 128 engage in individual transactions with the distributor node 124 for products that the distributor node obtains via the seller node 120, the central transaction node 110 facilitates payment between all parties. For example, payment from financial institutions for all three buyer nodes is carried out directly to the seller node 120's financial institution, with a (markup/service charge) portion of each payment being directed to the distributor node 124's financial institution. While the seller node 120 may simply be contracting with the distributor node 124, payment for the transaction may come directly from multiple financial institutions serving the buyer nodes 122, 126 and 128.

Multiple sellers may be implemented, similarly to the above-discussed approach with multiple buyers, for facilitating a single transaction between the distributor node 124 and the buyer node 122. In this instance, nodes 126 and 128 are additional seller nodes. The distributor node 124 contracts individually with each seller node 120, 126 and 128 for goods and/or services. When the buyer node 122 contracts with the distributor node 124 for the goods and/or services, the distributor node uses one or more of the seller nodes 120, 126 and 128 to fulfill the contract. The central transaction node 110 facilitates the transfer of funds between the buyer node 122 and the seller nodes 120, 126 and 128 as a function of the amount of goods and/or services provided by each seller node. For example, when the transaction involves products, the amount of product that is provided by each seller node is used to determine the amount that the seller node is to be paid. Similarly, when the transaction involves a service such as shipping, the amount of service performed (*e.g.*, corresponding to shipping distance, utility usage or other service work performed) is used to determine the amount that the seller is to be paid. The funds transfer is carried out using price-representing data stored in the database 112 that corresponds to goods and/or services

of the transaction as agreed upon between the distributor node 124 and the seller nodes 120, 126 and 128.

In each of the above-discussed examples involving a distributor, the transfer of funds between the parties to the transaction is carried out using one or more of the approaches discussed herein, such as those directed to monitoring and processing resources available for payment. When changes to order fulfillment and/or related payment(s) occur, these changes are used to automatically update status conditions. These changes, status conditions and other monitored characteristics can be implemented by the central transaction node 110 and used as a basis upon which to characterize and process payment for a transaction. For instance, contract performance (fulfillment), or the level of contract performance, can be used by the central transaction node 110 to authorize payment for the transaction.

In one implementation, the above-discussed transaction approach involving a mid-level entity is further implemented individually among sellers and buyers. For instance, payment for goods and/or services provided to a particular buyer is based upon fulfillment to that buyer. A particular buyer's financial institution is accessed for funds transfer based on quality, quantity and/or other conditions of transaction fulfillment. Similarly, where multiple sellers are used for a particular transaction, payment to each seller is based upon the level of transaction fulfillment provided by the seller.

In another example embodiment of the present invention, the central transaction node 110 acts as the mid-level entity as discussed in one or more of the above examples, wherein buyer and seller nodes (of nodes 120-128) interact with the central transaction node 110 for contracting purposes. In this regard, the central transaction node (or a corresponding portion thereof) is managed by a mid-level type entity, either via internal or external control by the mid-level entity. In some instances, buyers and sellers contract for prices directly with the central transaction node 110. In other instances, buyers and sellers contract with each other for setting prices, and one or more of a buyer and seller party to a transaction contract further with the central transaction node 110 for facilitating the transaction. For example, a seller may contract with the central transaction node 110 to manage the seller's distribution, financial processes or other transaction-based functions, such that the buyer contracts with the seller directly but processes transactions via the central transaction node 110. Sellers interacting with the central transaction node 110 are

effectively paid directly from buyers interacting with the central transaction node 110 for the sellers' goods and/or services. However, the central transaction node 110 (or mid-level entity functioning therewith) is paid a markup and/or a transaction fee for each transaction that it processes.

5 In various applications, including one or more of the above-discussed approaches discussed in connection with FIG. 1, a claims processor 130, claims data update controller 132 and/or a payment processor 134 are selectively implemented to respectively carry out transaction fulfillment processing, data field updates and payment facilitation. For example, where a seller 122 involved in a transaction between the seller and a buyer 120 submits a
10 claim (*i.e.*, in the form of an invoice) requesting payment against a particular transaction, the claims processor 130 is implemented to process the claim in a manner not inconsistent with that discussed above, to ascertain whether the claim is payable. That is, the claims processor 130 implements transaction-profile attributes and/or other information in the
15 database 112 (or in a distributed database 113) indicative of conditions that would facilitate or deny payment. For example, where the buyer 120 has updated information indicating that goods for which such a claim is made have been received, the claims data update controller 132 updates related historical fulfillment information in the database 112 that can be used in authorizing payment for the claim. That is, historical fulfillment data is audited to ensure that it reflects information in the receipt data and, where such receipt data has not
20 been updated to the historical fulfillment data, the claims data update controller 132 effects an update to the historical fulfillment data. The claims processor 130 uses the updated information to authorize payment (where such receipt is a condition upon which payment can be authorized, *i.e.*, the payment is also audited and validated as a function of the receipt/fulfillment data). When payment is authorized, the payment processor 134 uses the
25 authorization to facilitate payment to the seller 122 using, *e.g.*, financial entity 124 and/or other financial entities to draw funds from, extend credit on behalf of, or otherwise facilitate payment from the buyer 120.

 In some applications, the claims data update controller 132 monitors data in the database 112, in a distributed database 113 and/or at a user location such as a database
30 implemented by the buyer 120 or the seller 122 (with appropriate access granted to the claims data update controller by the buyer or seller). When changes to data fields occur, the claims data update controller 132 implements transaction-profile attributes, user profiles or

other information in the database 112 to determine whether the changes apply to related data fields. If such changes do apply, the claims data update controller implements such changes by updating the related data fields.

For example, when a buyer updates the fulfillment status of an order in response to receiving goods from a seller, the claims data update controller 132 uses such an update to update related fields as identified by the buyer, seller or other transaction party. Related data fields may include, for example, a data field specifying a payable status for a particular order in the database 112, or a fulfillment field for the order at a corresponding seller's database. Once such related fields are updated, the payment processor 134 facilitates payment as discussed above, where appropriate.

FIG. 2A is a flow diagram for a transaction processing approach, according to another example embodiment of the present invention. At block 210, business rules are established for processing transaction characteristics. These business rules include one or more of a variety of rules that direct the processing of transaction data for generating a particular result, such as the generation of information that can be used to further a transaction. At block 220, select data locations for data defined by the business rules are identified for having a relationship with other data locations. In some applications, data locations are automatically linked for a transaction as a function of parties to the transaction, with data fields corresponding to similar or the same data for different transaction parties being automatically linked. In other applications, data fields for a particular transaction party are linked, such as by linking a specific receivable field for a particular transaction to a general receivable field that characterizes all receivables for the transaction party.

If data stored in a data location having a relationship with other data locations is made at block 230, business rules that apply to the data location exhibiting a change are identified (*e.g.*, recalled) at block 240. The identified business rules are used in the automatic generation of updated information for related data locations at block 250. For instance, as discussed above, the generated information may include updated information for a transaction data location that is related to another location, with a change in a first data location being used with the business rules to execute a change in another data location. In another instance, the update to related data locations at block 250 involves the generation of data used to execute a transaction function, such as portion of a chain of events used for authorizing payment for a particular transaction. In the context of an invoice, changes in

quantities ordered, paid for and delivered can be identified and processed using this approach.

FIG. 2B is another flow diagram showing an approach for processing an invoice for a transaction between a buyer and a seller, according to another example embodiment of the present invention. At block 205, an invoice is received by the buyer and processed to determine whether the invoice applies to an order belonging to the buyer. If the invoice does not apply to an order belonging to the buyer at block 215, the process stops at block 225, with an output indicating the failure optionally generated (*e.g.*, a notice sent to the seller who created the invoice). If the invoice applies to an order belonging to the buyer at block 215, the invoice is parsed at block 235 to identify the quantity of a particular item invoiced. The identified quantity of a particular item is compared to related buyer resources at block 245 to determine whether the identified quantity being billed exceeds the amount fulfilled less the amount already invoiced (*e.g.*, on a previous invoice). If the identified quantity is greater than the difference between the amount fulfilled and the amount previously invoiced (if applicable), the quantity is over-billed and the process stops at block 225 (*e.g.*, with an optional output generated for notifying the seller of an over-billing situation and/or for updating related data fields). In one instance wherein the identified quantity is greater than the difference between the amount fulfilled and the amount previously invoiced at block 245, the quantity is adjusted to be equal to that difference (thus adjusted to pay for the total amount fulfilled), with the process continuing as follows. If the identified quantity is less than or equal to the difference between the amount fulfilled and the amount previously invoiced (if applicable), the identified quantity is determined to be payable at block 255.

Once the identified quantity is determined to be payable, the quantity is checked to ensure it meets a minimum payable quantity threshold at block 265. For example, a buyer may wish to limit the payment of invoices to those exceeding a particular dollar amount to reduce costs associated with paying invoices. This approach is applicable, *e.g.*, where payment for a few relatively inexpensive items of a large order would amount to only a few dollars; such payment would be withheld until an additional invoice is received. If the minimum payable quantity threshold is met (or if a minimum does not exist) at block 265, related fields are adjusted at block 285 and the portion of the received invoice that applied to the identified quantity is paid at block 295.

If the minimum payable quantity threshold is not met at block 265, but the identified quantity amounts to the total quantity of the order at block 275, the process continues at block 285 as discussed above. This instance is applicable, for example, to implementations where the invoiced quantity is small but the order is small, or when previous invoices amounting to nearly the total quantity have been paid, leaving only a few items unpaid. If the minimum payable quantity threshold is not met at block 265 and the identified quantity does not amount to the total quantity of the order at block 275, payment is held at block 276.

The adjustment of related fields at block 285 is optional and involves one or more of a variety of adjustments. For example, fields such as those relating to extended prices on a particular invoice at the line item level, service charges, taxes and other items. Moreover, items such as service charges and taxes can be processed similarly as shown in FIG. 2B as the "item" at block 245 (*e.g.*, with comparisons to past invoices for determining whether service charges, taxes or other items are appropriate).

FIG. 3 is a flow diagram showing an approach for categorizing data as a function of an anchor and monitoring data stored in locations corresponding to the anchor for automatically updating transaction processing data, according to another example embodiment of the present invention. The flow diagram in FIG. 3 may, for example, be implemented in connection with the flow diagrams shown in FIGs. 2A-2B and discussed above. For instance, the business rules mentioned in connection with FIG. 2A may be implemented in connection with the anchor profiles discussed further below.

At block 310 of FIG. 3, transaction data is sent to a central processor that is programmed for accessing, manipulating and using stored transaction-related data including anchor profile data, as well as for communicating with parties to a transaction or transactions to which the transaction data relates. The stored transaction-related data includes data stored under an anchor identification (ID) that is used to identify a particular transaction for which the transaction-related data is implemented. At block 320, the transaction data is parsed and compared with stored anchor profiles to detect and identify information in the anchor profiles that is consistent with (matches) or otherwise relates to the transaction data. If a match between the transaction data and a stored anchor profile is not found at block 330, a new matching anchor ID is created at block 340 for use in identifying the transaction data sent at block 310 with a particular transaction. If a match

between the transaction data and a stored anchor profile is found at block 330 (or, after a new matching anchor ID is created at block 340), selected elements of the transaction data are stored in the database under the matching (or new) anchor ID at block 350. These selected elements may, for example, be used to establish a new anchor profile either alone
5 or in connection with other transaction data (e.g., in the database or received at block 310 at a later time).

Once the transaction data is stored at block 350, business rules for the anchor ID corresponding to the transaction data are identified at block 360. These business rules may, for example, be part of the transaction data received at block 310. In another example, these
10 business rules are pre-stored under a particular user or anchor ID to which the transaction data relates. When the relationship between the transaction data and the user or anchor ID for particular business rules is established, the central processor then uses the particular business rules to monitor and process the transaction data. Specifically, transaction data is monitored in accordance with the identified business rules at block 370 and when a change
15 in transaction data (or new transaction data) is detected, related transaction data under the anchor ID is updated at block 380 in accordance with the business rules. Alternatively, when the change in transaction data applies to data stored under more than one anchor ID (e.g., as discussed below with the scenario with two invoices for a particular order), transaction data under another anchor ID is also updated. In either instance involving a
20 change in transaction data (or, in the instance where the received document is an invoice, involving a proposed change in transaction data, namely, a payable quantity), the approaches discussed in FIGs. 2A-B can be implemented here.

In one particular example embodiment of the present invention, the following data attributes, documents and identifications are implemented by a central processor to group
25 transactions that are common to a particular interaction between buyer and seller parties. This approach may be implemented, for example, in connection with the data storage and user nodes as shown and discussed above in connection with the transaction processing arrangement 100 shown in FIG. 1. The central transaction node 110 manages sets of information about an interaction as programmed by applicable business rules configuration
30 that can be set at any combination of a variety of rules. For example, programmed business rules may be used in connection with sets of information related to the fulfillment status of an order as discussed in connection with Table 1 below.

Referring below to Table 1 and to FIG. 1, transaction attributes are shown for two documents in connection with a transaction between a buyer and seller as processed by the transaction processing arrangement 100, according to another particular example embodiment of the present invention. When a first document received by the central transaction node 110 is Invoice "123" from Seller "S1" naming Order "ABC" from Buyer "B1" as the justification for the billing, the central processor responds by searching and matching the invoice to information stored in the database. Based on the search (*e.g.*, a logical key search or an algorithm-based search), the central transaction node 110 determines whether it has any previous knowledge of an interaction with the unique attributes used for transaction document identification purposes, here being attributes "123," "S1," "ABC" and "B1." If previous knowledge of the transaction to which the invoice applies is found, the central transaction node 110 aligns the attributes from the invoice with known attribute field, such as that from Receipt "R789" and Order "ABC."

Once the invoice has been associated with other documents/information, the central transaction node 110 performs a variety of transaction-related operations, such as those discussed above, using data from the Invoice "123" and other documents and information in the database 112. For example, when a quantity of a product or other transaction-related item ordered for the transaction is identified, the Ordered Not Billed quantity resource "QTY1" at the line item level is increased for the Receipt "R789." The Ordered Not Received quantity resource for Receipt "R789" is also correspondingly updated to match the increase in the Ordered not Billed quantity. Similarly, when the quantity of a product or other transaction-related item ordered for the transaction is identified, the Ordered Not Received quantity resource "QTY2" at the line item level is increased for the Invoice "123." In this discussion and as otherwise applicable to one or more example embodiments herein, the line item level refers generally to line items, within a particular document, that specify information particular to a line item, where several line items are included in a single document.

The cost of a particular quantity for a transaction is added to either the Ordered Not Billed monetary resource "AMT1" at the line item level or the Ordered Not Billed monetary resource "AMT2" at the document level. This approach is useful, for example, when the total price of an order is different than that for a particular invoice or receipt for a portion of

the order (*i.e.*, with the document level referencing an entire order amount, rather than merely the amount for a particular invoice).

When goods are received before Invoice "123," the Received Not Billed quantity resource "QTY3" at the line item level is increased. The total quantity of Order "ABC" at the document level may also be increased in the amount that the quantity resource "QTY3" is increased, which may also correspondingly apply to the Ordered Not Billed and Ordered Not Received quantities at the line item level.

When Invoice "123" is received, the amount billed subtracts from the Ordered Not Billed quantity resource "QTY1" at the line item level as well as the Received Not Billed quantity resource "QTY3" at the line item level. In addition, the amount billed subtracts from either the Ordered Not Billed monetary resource "AMT1" at the line item level or the Ordered Not Billed monetary resource "AMT 2" at the document level. Moreover, as additional invoices are received for Order "ABC," the corresponding quantity resources are continually updated to provide accurate information.

With the approaches discussed above in connection with Table 1 below, the effect of backing out a source of resources can be immediately rolled through all the financial documents claiming resources that have not yet been paid. This technological approach has been found useful in effectively automating the document pre-payment reconciliation process for physical and financial fulfillment.

ATTRIBUTE	Invoice 123	Receipt R789	Order ABC
AnchorID	"A1"	"A1"	"A1"
Named Buyer	"B1"	"B1"	"B1"
Named Seller	"S1"		
Named Order	"ABC"	"ABC"	"ABC"
Named Invoice	"123"		
Source Document Owner	"S1"	"B1"	"B1"
Source Document Owner Type	Seller	Buyer	Buyer
Source Document Type	Invoice	Receipt	Order
Source Document ID	"123"	"R789"	"ABC"
Ordered Not Billed quantity (line item level)		"QTY1"	"QTY1"
Ordered Not Rec'd quantity (line item level)	"QTY2"		"QTY2"
Ordered Not Billed monetary (line item level)		"AMT1"	"AMT1"
Ordered Not Billed monetary (doc. level)			"AMT2"
Received Not Billed (line item level)		"QTY3"	"QTY3"

Table 1

FIG. 4 shows an arrangement for processing claims for a plurality of transactions, according to another example embodiment of the present invention. Various ones of the example embodiments and implementations discussed herein are selectively implemented using the arrangement in FIG. 4. An example application of the arrangement in FIG. 4 involving a buyer and a seller is discussed as follows; while described in connection with this example, the arrangement in FIG. 4 is configured for processing similar transactions for a multitude of disparate buyer and seller combinations, using transaction-profile attributes pertaining to transactions involving each buyer and seller combination.

10 A seller 402 generates an invoice 405 and sends the invoice to a transaction processor 410. In response to receipt of the invoice 405, an association processor 412 associates the invoice with a particular transaction and sends a transaction profile request 430 to a databank arrangement 440, which returns transaction profiles 431. An update processor 414 uses the returned transaction profiles to identify an appropriate historical billed quantity 444 (*e.g.*, a data field) in the databank arrangement 440, which is updated with historical billed quantity update information 432.

When a buyer 420 generates a receipt 425, the association processor 412 associates the receipt with a particular transaction and sends another transaction profile request 430 to the databank arrangement 440, which returns appropriate transaction profiles 431. The update processor 414 uses the transaction-profile attributes and the receipt data to update an appropriate historical fulfillment quantity 446 (*e.g.*, a data field) in the databank arrangement 440.

The transaction processor 410 is adapted to process in accordance with the above-discussed approaches involving the receipt of the invoice data 405 and the receipt data 425 in different orders. Regardless of the order, the claims processor 416 is responsive to the invoice 405 by sending an historical data request 433 to the databank arrangement 440, which returns historical billed and fulfillment quantity data 434. The claims processor 416 uses the returned historical billed and fulfillment quantity data 434 to determine whether payment for some or all of the invoice 405 is appropriate. For instance, when the historical fulfillment quantity exceeds the historical billed (*e.g.*, and paid) quantity by an amount equal to a quantity billed in the invoice 405, the claims processor 416 determines that the entire invoice can be paid.

Once payment is authorized by the claims processor 416, a payment processor 418 uses the authorization to facilitate payment for the invoice 405 by sending a payment authorization 411 to a financial institution 450 identified, *e.g.*, in the returned transaction-profiles 431. The financial institution 450 in turn sends payment 451 to the seller 402.

5 While certain aspects of the present invention have been described with reference to several particular example embodiments, those skilled in the art will recognize that many changes may be made thereto without departing from the spirit and scope of the present invention, aspects of which are set forth in the following claims.

WHAT IS CLAIMED IS:

1. A transaction-processing system for managing payment resources for a plurality of transactions involving merchant offerings among parties including buyers and sellers, different ones of the plurality of transactions involving different sets of buyer and seller parties, the system comprising:

a data storage arrangement adapted, for each of the plurality of transactions, to store data characterizing historical fulfillment and billed quantities for ordered items of the transaction; and

a computer arrangement adapted, for each of the ordered items listed on an invoice for each of the plurality of transactions, to

associate the invoice with one of the plurality of transactions;

authorize payment for each ordered item listed on the invoice as a function of a billed quantity of the ordered item on the invoice and a difference between the historical fulfillment quantity and the historical billed quantity for the ordered item as indicated in the data stored in the data storage arrangement for the associated transaction, and

update historical billed quantities for the ordered item as characterized in the data stored in the data storage arrangement as a function of the billed quantity in the invoice.

2. The system of claim 1, wherein the computer arrangement is adapted to, in response to the billed quantity of the ordered item on the invoice being more than the difference between the historical fulfillment quantity and billed quantity information for the ordered item in the data storage arrangement, identifying the quantity of the particular ordered item on the invoice that exceeds the difference between the historical fulfillment quantity and billed quantity information as over-billed.

3. The system of claim 2, wherein the computer arrangement is adapted to authorize payment for a quantity of the particular ordered item on the invoice that does not exceed the difference between the historical fulfillment quantity and billed quantity information for the ordered item in the data storage arrangement.

4. The system of claim 3, wherein the computer arrangement is adapted to communicate the quantity of over-billed items on the invoice to the seller party to which the over-billed items apply.
5. The system of claim 1, wherein the computer arrangement is adapted to receive information confirming fulfillment of a particular item for an order and to automatically update the historical fulfillment quantity data in the data storage arrangement for the particular item to reflect the confirmed fulfillment information.
6. The system of claim 1, wherein the computer arrangement is adapted to:
monitor changes made to selected data fields in the data storage arrangement; and
in response to a change in one of the selected data fields, update another related data field.
7. The system of claim 6, wherein the computer arrangement is adapted to automatically link related data fields for a particular transaction as a function of a transaction party to which the related data fields apply.
8. The system of claim 1, wherein the computer arrangement is adapted to compare information in each invoice with profile information stored in the databank to match the invoice with historical fulfillment and billed quantities and to use the matched historical fulfillment and billed quantities when authorizing payment for each ordered item listed on the invoice.
9. The system of claim 1, wherein the data storage arrangement is adapted to store transaction-profile attributes for each transaction, the transaction-profile attributes including information for identifying order information for the transaction and at least one of a buyer and a seller involved with the transaction, wherein the computer arrangement is adapted to authorize payment for each ordered item listed on the invoice as a function of the transaction-profile attributes.

10. The system of claim 9, wherein the computer arrangement is adapted to monitor changes in a transaction-profile attribute that indicates the fulfillment status of an order and, in response to changes in the fulfillment status of a particular order, update other transaction-profile attributes that relate to the fulfillment status for the particular order.

11. The system of claim 10, wherein the computer arrangement is adapted to update transaction-profile attributes including payment-related attributes in response to a particular order being fulfilled.

12. The system of claim 10, wherein the computer arrangement is adapted to facilitate payment for the particular order in response to an update to payment-related attributes for the particular order indicating that payment is appropriate.

13. The system of claim 1, wherein the computer arrangement is adapted to authorize payment for each ordered item listed on the invoice as a function of a billed quantity of the ordered item on the invoice and a difference between the historical fulfillment quantity and the historical billed quantity for the ordered item as indicated in the data stored in the data storage arrangement by comparing billed quantity information for the ordered item on the invoice with the historical fulfillment and billed quantity information for the ordered item as indicated in the data stored in the data storage arrangement.

14. The system of claim 13, wherein the data storage arrangement is adapted, for each of the plurality of transactions, to store data characterizing a paid quantity for ordered items of the transaction and wherein the computer arrangement is adapted, in response to payment being authorized, to update the data characterizing the paid quantity for the ordered items in the data storage arrangement.

15. The system of claim 1, wherein the computer arrangement is adapted to audit and validate the stored data characterizing historical fulfillment quantities as a function of receipt data, thereby validating that received quantity resources have been added to the historical fulfillment quantities.

16. The system of claim 1, wherein the data storage arrangement is adapted to store data characterizing historical fulfillment and billed quantities for ordered items of a the transaction on a line item by line item basis and wherein the computer arrangement is adapted to authorize said payment for and update said historical billed quantities on a line item by line item basis.

17. The system of claim 1, wherein the computer arrangement is adapted to update historical fulfillment data in the data storage arrangement as a function of the time of delivery of an ordered item.

18. The system of claim 1, wherein the computer arrangement is adapted to update historical fulfillment data in the data storage arrangement as a function of the condition of the ordered item upon delivery thereof.

19. A method for automatically managing payment resources for a plurality of transactions involving merchant offerings among parties including buyers and sellers, the method comprising, for each of the plurality of transactions:

associating invoice data with a particular one of the transactions as a function of data in the invoice and stored transaction-profile information; and

authorizing payment from a buyer to a seller for seller-invoiced item quantities as a function of

fulfilled quantity data indicating items that have been fulfilled by the seller for the transaction, and

historical billed status data for the item, the historical billed status data being indicative of a portion of the total item quantity that has been previously-billed.

20. The method of claim 19, wherein authorizing payment includes authorizing payment for at least a portion of the seller-invoiced item quantities that does not exceed a difference between the fulfilled quantity data and previously-billed item quantity.

21. The method of claim 20, wherein authorizing payment includes authorizing payment for at least a portion of the fulfilled quantity of the seller-invoiced item quantities that does not exceed an order total quantity for the item.

22. The method of claim 19, further comprising:
in response to authorizing payment for seller-invoiced item quantities, storing data indicating the quantity of the item for which payment was authorized.

23. The method of claim 19, further comprising:
in response to receiving item quantity receipt data from a buyer for a particular transaction, associating the receipt data with a particular one of the transactions for seller-invoiced item quantities as a function of the transaction-profile information, and updating historical fulfillment data for the associated transaction indicating the quantity of the item indicated in the receipt data.

24. A method for managing payment resources for a plurality of transactions involving merchant offerings among parties including buyers and sellers, different ones of the plurality of transactions involving different sets of buyer and seller parties, the method comprising:

for each of the plurality of transactions, storing data characterizing historical fulfillment and billed quantities for ordered items of the transaction; and

for each of the ordered items listed on an invoice for each of the plurality of transactions,

associating the invoice with one of the plurality of transactions;

authorizing payment for each ordered item listed on the invoice as a function of a billed quantity of the ordered item on the invoice and a difference between the historical fulfillment quantity and the historical billed quantity for the ordered item as indicated in the data stored in the data storage arrangement for the associated transaction, and

updating historical billed quantities for the ordered item as characterized in the data stored in the data storage arrangement as a function of the billed quantity in the invoice.