

Table 18, query service provider Q_provider (e.g., 32) is associated with each service module.

TABLE 18

Example SCOL_QUERY Table					
SVC_MODULE_NAME*	QUERY_NAME*	RESULT_ASPECT	PARAM_STRUCTURE	INPUT_KEY_ASPECT	PROVIDER_CLASS
S_Customer	QueryByName	Customer_Key	Customer_Structure	Customer_Key	Q_provider
S_Product	QueryByName	Product_Key	Product_Structure	Product_Key	Q_provider
S_Basket	QueryByCustomer	Basket_Key	Customer_Key_Table	Customer_Key	Q_provider
S_OrderHeader	QueryByCustomer	OrderHeader_Key	Customer_Key_Table	Customer_Key	Q_provider

[0118] Table 19 defines a data dictionary for the implementation of repository 18 defined in Tables 13-18. Each row defines a structure having a name and multiple data entries and their types. For example, structure Customer_Structure has one data entry titled "NAME" with a CHAR type indicating a character string. The Customer_Key_Table structure defines a CustomerId key for each customer and also has a CHAR type.

TABLE 19

Example SCOL_STRUCT Table				
STRUCT_NAME*	DATA1	TYPE1	DATA2	TYPE2
Customer_Structure	NAME	CHAR		
Customer_Key_Table	CustomerId	CHAR		
Address_Structure	Town	CHAR	Street	CHAR
Address_Key_Table	AddressId	CHAR		
Product_Structure	Name	CHAR	ProductId	CHAR
Product_Key_Table	ProductId	CHAR		
Basket_Structure				
Basket_Key_Table	BasketId	CHAR		
Position_Structure				
Position_Key_Table	PositionId	CHAR		
OrderHeader_Structure	Submitted	CHAR		
OrderHeader_Key_Table	OrderId	CHAR		

[0119] The database operations described above for process 150 are defined in this implementation of repository 18 as follows. Querying (154), included in process 150, of the Customers database table (Table 7) is described in meta data repository 18 by the QueryByName query associated with aspect service module S_Customer in Table 18. This QueryByName query associated with aspect service module S_Customer is selected because the front end application program 12 has obtained a customer name and service module S_Customer contains aspects with customer names. For example, front end application program 12 can submit query QueryByName associated with service module S_Customer with NAME="John Smith" and no filtering Customer_Key specified to service manager 16. Service manager 16 checks repository 18 to ensure that the query is defined. Service manager 16 then submits the query to Q_provider (e.g., 32) that queries the Customer database

table (Table 7) in database 24 and the output is sent back to front end application program 12 is a record set containing CustomerId={1}.

[0120] Inserting (160), included in process 150, on Customers Addresses database table (Table 8) and inserting (162), included in process 150, on Customers database table (Table 7) are described by standard Insert operations (described previously) on aspects A_Address and A_Customer, respectively, in the meta data repository 18.

[0121] Querying (166), included in process 150, on the Products database table (Table 9) for a product name is described by QueryByName query associated with service module S_Product defined in Table 18. For example, application 12 can submit the query QueryByName associated with service module S_Product with Name="Wrench" and no filtering Product_Key specified to service manager 16. Service manager 16 checks repository 18 to ensure that the query is defined. Service manager 16 then submits the query to Q_provider (e.g., 32) queries database 24 and the output sent back to application 12 is a record set containing ProductId={3}.

[0122] Inserting (170, 172, and 174), included in process 150, are defined by insert operations on aspects A_OrderHeader, A_Basket, and A_Position, respectively.

[0123] Querying (178), included in process 150, Orders database table (Table 12) by customer is described by the QueryByCustomer query associated with service module S_Order defined in Table 18. For example, front end application program 12 can submit query QueryByCustomer associated with service module S_Order with Customer_Key (CustomerId)={2} and no filtering OrderHeader_Key. Service manager 16 checks repository 18 to ensure that the query is defined. Service manager 16 then submits the query to Q_provider (e.g., 32) that queries database 24 and the output is sent back to application 12 is a record set containing OrderHeader_Key (OrderId)={2, 3}.

[0124] Selecting (180), included in process 150, order operation on Orders database table (Table 12) and setting (182) submitted field to true on selected orders are defined by the Execute Submit action (defined in Table 14) on aspect A_OrderHeader in repository 18. For example, front end application program 12 sends the Submit action on aspect A_OrderHeader to service manager 16 with OrderHeader_Key={2, 3}. Service manager 16 then sends the submit