Tabular Application Development

Talib Damij and Janez Grad
University of Ljubljana, Faculty of Economics, 1000 Ljubljana, Slovenia

Keywords: MIS, object-oriented methods, systems analysis

Edited by: R. Murn
Received: January 15, 1996 **Revised:** July 18, 1996 **Accepted:** August 22, 1996

The aim of this work is to introduce an effective object-oriented approach which guides the analyst through all phases of the application development and puts the whole process under the complete control of the analyst.
Tabular application development is a method which develops the application by creating several tables. It has four phases and introduces a new idea for developement an application. The first phase defines the problem to be solved. The second phase deals with the analysis of the application. The third phase designs the application in detail, and the fourth phase deals with the implementation of the application.

1 Introduction

This work represents an effective approach called tabular application development (TAD). This approach defines an order in which the whole process of application development is put under the complete control of the analyst. TAD is a new approach, because it introduces a new idea for developing the application by creating several tables. It has four phases. The first phase is the problem definition which identifies the critical success factors, the objectives and the entities of the organization using the factor-objective and entity tables. The second phase is the analysis which defines the actions of the system using the action table, identifies the application's data structures, develops the object model and analyses the outputs of the application. The third phase is the design which uses the information gathered in the factor-objective, action, linkage tables and in the object model to create the application model. This phase also defines the algorithms and generates priority order values. The final phase is implementation which transfers the objects into classes and writes a code according to the algorithms.

The first phase of TAD defines the problem to be solved. The best way to do this is to organize interviews with all users. First we interview the management and then continue with other users. This phase has two steps: the first is to identify the management's critical success factors and objectives, and the second to identify the enteties of the organization.

2 Problem definition

This work will treat a simplified example of a sales organization.

2.1 Critical success factors and objectives

This step tries to identify the critical success factors and the objectives of management. The analyst may start with the second step if it is not possible to perform this stage.

Usually we start the process of interviewing with the top management and continue with the management at lower levels. The purpose of these interviews is to discover the management's expectations and to ensure that each of their requirements and objectives is considered.

The management has to define the critical success factors and the objectives of the organization. The analyst will first ask the top management to determine these factors. In most corporations there are a small number (3-6) of factors which are vital to the success of the organization [Martin, 1982]. When the top management has defined a list of critical success factors and a list of objectives they will be asked to define measurements which enable the application to control these factors and objectives. This means that for each critical success factor or objective the management has to define the information sources and the way in which these sources will be controlled. This process will be repeated with the management at different levels.

The result of these interviews is a list of critical success factors, a list of objectives, their information sources and the ways in which they will be realized. These results will be collected in a table called the factor-objective table.

In the example of a sales organization the following success factors are identified:

1. The organization has to assure a 5% growth per year.
2. The time from the acceptance of an order to the shipment of the ordered products must not be longer than one day.
3. Customers have to complete their payments in eight days.

The management also defines the following three objectives:

1. Permanent profit-making which enables the organization to invest in new technology.
2. Increasing the organization's market.
3. Increasing the organization's export.

These critical success factors and objectives are recorded in Table 1, which shows the factor-objective of the sales organization. An asterisk indicated in any square (i,j) in the factor-objective table means that the source defined in column j must generate information required by the critical success factor or the objective defined in the row i.

2.2 The entities

An entity is any source which is part of the system or is connected with the system by some interaction. So, an entity may be a user or any other source which receives an action (event) from the system or sends an action to the system.

The management helps in identifying the system's entities and creating a plan of inteviews with all users. This plan should be developed in accordance with the hierarchical view of the organization. In this way every user will be consulted and each of his/her requirements or objectives will be registered.

Identifying the entities, their requirements and actions will be achieved by developing a table called the entity table. This table is structured as follows: The rows and also the columns of the table represent the entities of the system. According to this idea every entity occupies one row and one column of the table.

The entity table is developed as a result of the interviews. These interviews should be organized only with the internal users. Internal users inform us about the behaviour of external users and other entities. The internal entity is a part of the system. The external entity is not a part of the system, but it has one or more interactions with the system. To create this table we usually start the process of interviews with the users as they are specified in the rows of the table. So, we start the interviews with the entity defined in the first row, if this entity is an internal user , and then move to the entity defined in the second row. Following this order we continue until we have dealt with all the rows.

Every entity creates a part of the entity table. Let us deal with the entity defined in row i, where i is from 1 up to the number of entities. For this entity we register all actions using the following rules:

1. The name(s) of the action(s) will be written in square (i,i) if the entity in row i creates some action for its use only.
2. The name(s) of the action(s) will be written in square (i,j) if the entity in row i sends this action to the entity defined in column j.

Table 2 illustrates the entities of the sales organization. The analyst identifies five entities. These are Customer, Salesman, Sales Department, Shipping Department and Accounts Department. From the interview with the internal users he found that each Salesman covers a determined number of customers. The Salesman visits his Customers from time to time and collects their orders. The Sales department recei-

Factor,Obj.	Source				Manage ment level	Result
	Order	Shipment	Invoice	Payment		
1. factor	*				Top	Report
2. factor	*	*			Top	Report
3. factor				*	Top	Report
1. objective	*		*	*	Top	Report
2. objective	*				Top	Report
3. objective	*				Top	Report

Table 1: The factor-objective table of the sales organization

ves these orders from Salesmen and sends a copy of these documents to the Shipping Department. For this reason he writes ORDER in the squares (1,2), (2,3) and (3,4). Furthermore, he found that the Shipping Department creates a shipment for every order received. One copy of the shipment is sent to the Customer and an other copy to the Accounts Department. So, he registers SHIPMENT in squares (4,1) and (4,5). Finally he finds that the Accounts Department creates an invoice and sends it to the Customer and then receives a payment from the Customer. For this reason he writes INVOICE in square (5,1) and PAYMENT in square (1,5).

Table 2 also shows that the actions in the entity table are numbered as they happened in the real world.

The result of the problem definition phase is identifying the strategic or business objectives of the management, the critical success factors of the organization, the system's entities and their actions.

3 The analysis

This phase in TAD has five steps. The first step deals with the actions of the entities. The second step defines the data structures of the application. The third step identifies the application's objects and their relationships. The fourth step develops the object model. The last step deals with creating the outputs required by the management.

3.1 The actions

To identify the application's actions let us link every action defined in the entity table with the entities of this table. The best way to do this is by developing a table called the action table. Creating this table leads to discovery of the whole application system, its subsystems and its actions.

The action table is organized as follows: The actions of the system are represented by the rows and the entities are represented by the columns.

To create this table we list the actions in the same order as they occur in the real world. So, the first action occurring (event) will be defined in the first row, second action in the second row and so on. For every action defined in row i, where i is from 1 up to the number of actions, we list the entities in the columns and try to link this action with each of these entities. If any connection exists between action i and entity j, where j is from 1 up to the number of internal entities, then an asterisk is written in square (i,j).

Table 3 shows the action table of the sales organization. From this table we can see that the complete application system is represented by the whole table. Furthermore any column of the table gives a clear picture of the connections of the entity defined in this column and the actions in the rows which are indicated by letters S or T. Letter S indicates a source entity and letter T means a target entity. So, each column represents a determined subsystem. Every subsystem includes one or more actions. On the other hand every row of the table shows exactly what is happening with the action defined in this row.

Hence the Table 3 shows the action table of the sales organization.

3.2 Data structures

The second step of the analysis transforms the user's models into stable data structures. This will be achieved by using the normalization te-

Entity	Customer	Salesman	Sales Department	Shipping Department	Accounts Depatrment
Customer		1 ORDER			4 PAYMENT
Salesman			ORDER		
Sales Department				ORDER	
Shipping Depatrment	2 SHIPMENT				SHIPMENT
Accounts Depatrment	3 INVOICE				

Table 2: The entity table of the sales organization

Action	Customer	Salesman	Sales Department	Shipping Department	Accounts Depatrment
1. Order	S1	T1, S2	T2, S3	T3	
2. Shipment				S	T
3. Invoice	T				S
4. Payment	S				T

Table 3: The action table of the sales organization

chnique. The result of this step is a set of normalized relations. The following relations are the normalized relations of the sale organization.

- Order (Order#, Cust#, Date)
- Shipment (Ship#, Cust#, Order#, Date)
- Invoice (Inv#, Cust#, Ship#, Date, Value)
- Payment(Pay#,Cust#,Inv#,Date,Value)
- Customer (Cust#, Name, Address)
- Product (Product#, description, Price)
- Order-Product (Order#,Product#, Qty)
- Ship-Product (Ship#,Product#, Qty)

3.3 The objects

The process of identifying the objects has two steps. The first step defines different groups of objects and the second step merges these groups together in a unique group.

An object is any thing, real or abstract, about which we store data and those operations that manipulate the data [Martin,1993]. Corresponding to this definition and to the TAD methodology we may conclude that an object is any entity, action and data structure about which the user stores information.

For this reason we define three groups of objects. The first group is obtained by analyzing the entities which are presented in the entity table. If the user collects information about any of these entities then we transform this entity into an object.

In the example of the sales organization we found five entities (Table 2). These are Customer, Salesman, Sales Department, Shipping Department d anthe Accounts Department. Furthermore, the organization stores information about Customers and Salesmen. So, the first group of objects contains the following two objects: Customer, Salesman.

The second group of objects is gained by transforming every action listed in the action table into an object. In our example the action table (Table 3) represents four actions. These actions we transform into objects. For this reason the second group has the following four objects: Order, Shipment, Invoice and Payment.

The third group of objects is found by transforming the data structures into an objects. In our we have eight data structures defined in 3.2. So, in the third group of objects we define eight objects. These are: Order, Shipment, Invoice, Payment, Customer, Product, Order-Product and Ship-Product.

The second step of identifying the objects deals with merging these three groups of objects together. The result of this step is a unique group of objects about which the user wants to store data and operations which manipulate this data.

In the example of the sales organization we merge the above defined three groups to a single group which contains the following objects: Order, Shipment, Invoice, Payment, Customer, Salesman, Product, Order-Product and Ship-Product.

3.4 The object model

From the previous step we obtain a set of objects. To develop the object model we need to identify the attributes of these objects, their associations and their operations. So, the process of creating the object model has three steps.

The first step deals with analyzing every object and identifying its attributes if these attributes have not yet been identified.

In our example we find that the only object which needs further analysis to identifiy its attributes is Salesman. The attributes of other objects were defined in 3.2. From the problem definition phase we find that the organization wants to obtain information about the sales realization of every salesman. Moreover we find that every salesman covers a determined number of customers. This means that every customer has a determined salesman. For this reason we connect the object Salesman with attributes such as Identity number and Name. To link the salesman with his customers we extend the object Customer by one attribute, and this is the identity number of the salesman.

The second step analyses the associations between the objects. The best way to do this is by developing a table called the linkage table. This table is structured as follows: all objects are represented in the rows and also in the columns in the same order. So, every object occupies one row and one column with the same index value. In the linkage table we try to link the objects listed in the rows with the objects listed in the columns. Links existing between objects may be identified successfully by searching for foreign and part keys.

To this purpose we create the linkage table by listing the objects defined in the rows. For every object in the row i, where i is from 1 up to m and m is the number of objects, we list all objects defined in the columns. If object(i) contains a key attribute or part of the key attribute of object(j) in the columns, where j is from 1 up to m, then a letter (K, F or P) will be written in square (i,j).

Letter K (key) means that object(i) and object(j) have a key attribute. Letter F (foreign key) means that object(i) contains a key attribute of object(j). Letter P (part key) means that the key attribute of object(i) is part of the key attribute of object(j).

For these reasons we may say that: letter K indicates an Isa structure, letter F incadites an Is-associated-with structure and letter P indicates an Is-part-of or Is-asociated-with structure.

Table 4 shows the result of implementing this procedure in our example.

The third step uses the information in the linkage table to develop the object model of the application and to identify the operations of each object. To do this we create the following procedure:

```
for every object(i) defined in the rows of the linkage table, where i = 1 to m
  if object(i) is not drawn yet then
    draw object(i), write its attributes and identify the operations which manipulate its data.
  for every object(j) in the columns of the linkage table, where j = 1 to m
    if object(j) is not drawn yet then
      draw object(j), write its attributes and identify the operations which manipulate its data.
    if square(i,j) is not empty then
      connect object(i) to object(j)
      add to object(i) the needed operations to manipulate the data of object(j).
```

Implementing the above defined procedure results in the object model listed in Figure 1.

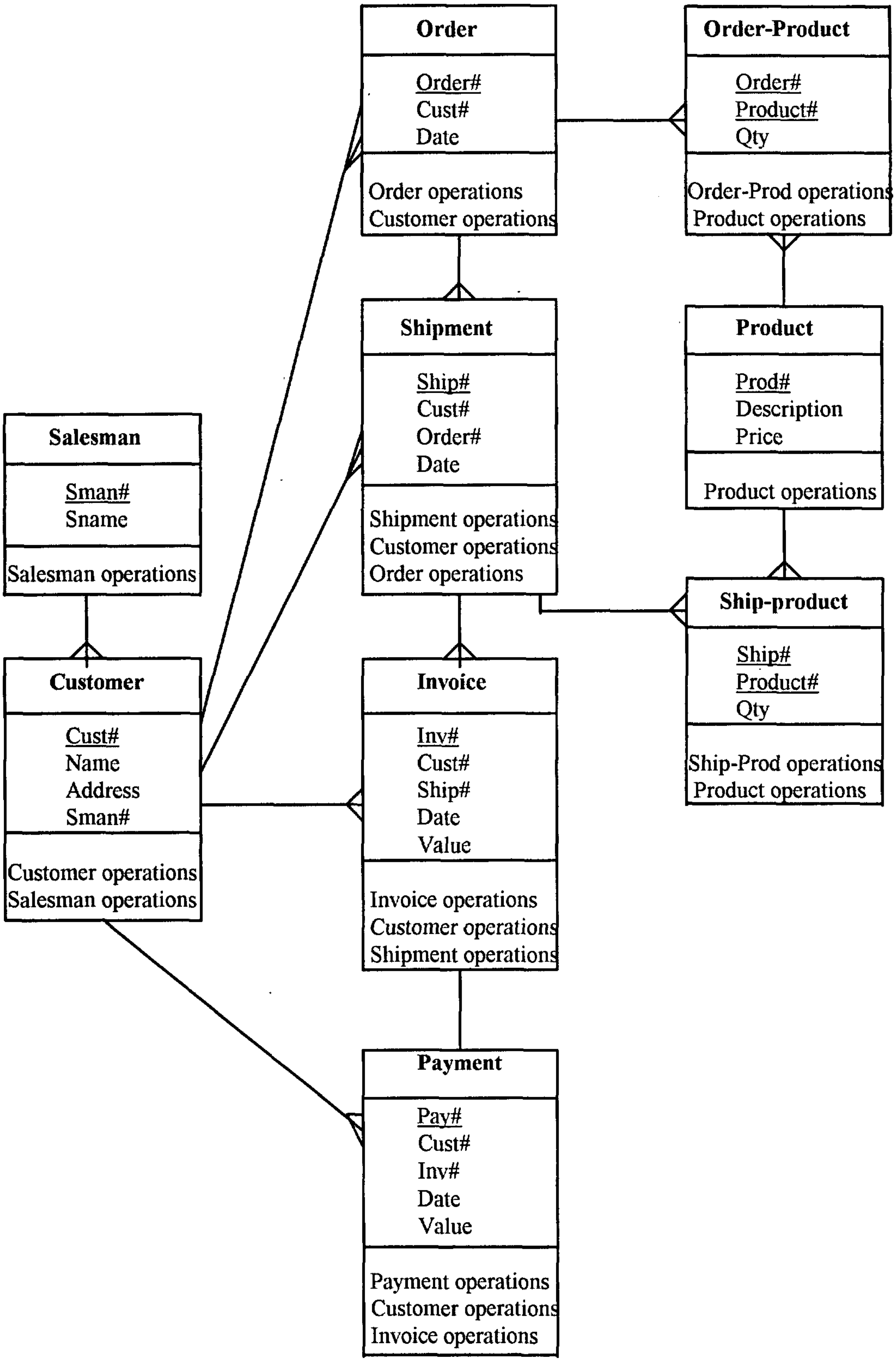

Figure 1: The object model of the sales organization

Object	Order	Shipment	Invoice	Payment	Customer	Salesman	Product	Order-Product	Ship-Product
Order					F			P	
Shipment	F				F				P
Invoice		F			F				
Payment			F		F				
Customer						F			
Salesman									
Product								P	P
Order-Prod									
Ship-Prod									

Table 4: Linkage table of the sales organization

3.5 The outputs

This step deals with analyzing and defining the outputs of the application. We try to find out if the object model defined (Figure 1) enables us to create all the outputs expected by the management and other users.

In this step we consider particularly the outputs which are required by the critical success factors and the objectives defined in the factor-object table.

If one or more outputs cannot be generated from the existing data, then we have to return to the users for more information to eleminate this incompleteness. This means that we have to return to the problem definition phase for greater clarity and accuracy in the definition of the user's requirements.

From this we can conclude that TAD is an iterative process. With every new iteration more problems may be solved and clarification achieved, and the system made more complete.

4 The design

This phase has three steps. The first step develops the application model, the second step writes an algorithm for each action and the third step defines the order of implementation the application model.

4.1 The application model

The first step of the design develops a model of the application using the information existing in the action, and factor-objective tables.

According to the action table we may see that the complete application system is represented by the whole table. Each column of this table is occupied by an entity, which means that every column represents a subsystem. Each subsystem contains one or more actions which are indicated by the asterisks in this column. For this reason we may say that the action table enables us to create the model of the application with all its subsystems and actions.

The action table is very valuable for determining convenient entity access to the data. So, every entity in any column may access those actions (objects) which are indicated by asterisks in that column.

In addition to this, we have to extend the created model of the application by the required outputs. Information about these outputs is contained in the factor-objective table. These outputs are very important to management and may guarantee their support.

Figure 2 shows the application model for the sales organization.

4.2 The algorithm

The second step of the design deals with creating the data flow diagrams and the algorithms of the application. According to the application model the analyst creates a DFD and writes an algorithm for every defined action. To do this we use the information existing in the application model, the object model, the factor-objective table and the action table.

- Sales organization
  - Management
    - Reports
  - Salesman
    - Sale realizaton
  - Sales Department
    - Reports
    - Create Order
  - Shipping Department
    - Reports
    - ReadOrder
    - Create Shipment
  - Accounts Department
    - Reports
    - Read Shipment
    - Create Invoice
    - Create Payment

Figure 2: The application model of the sales organization

5 The implementation

In this phase the analyst converts the objects into classes and the operations into methods. Furthermore, we generate a code according to the method specifications and to the defined data flow diagrams and algorithms.

The analyst may also define a convenient data access for every user corresponding to information in the action table and to the model of the application.

6 Conclusion

The aim of this work was to introduce an effective object-oriented approach which guides the analyst through all phases of the application development. This approach puts the whole process of application development under the complete control of the analyst. For this reason it is independent of the analyst and his/her experience.

Developing an application using TAD minimizes the time needed to complete this process and puts the application under the full control of the developer the whole time. These characteristics to make this approach very acceptable and useful in practice.

References

[1] T. DAMIJ, An Approach for Information Systems Development. Proceeding of The International Conference on Information Systems Development - ISD '94, Methods & Tools & Theory & Practice, University of Maribor, School of Organizational Scienes, Kranj, 1994, str. 173-182.

[2] T.DAMIJ, Tabular Based Approach for Systems Development. Proceeding of The International Conference on Organization and Information Systems, University of Maribor, School of Organizational Scienes, Kranj, 1995, str. 457-461.

[3] J. MARTIN, Strategic Data-Planning Methodologies. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1982.

[4] J. MARTIN, Principles of Object-Oriented Analysis and Design. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1993.

[5] G. L. SANDERS, Data Modeling. Boyd & Frase Publishing Company, Danvers, 1995.

[6] T.W. OLLE et al., Information Systems Design Methodologies: A Comparative Review. Elsevier Science Publishers B.V., North-Holland, 1982.

[7] J. RUMBAUGH et al., Object-oriented Modeling and Design. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1991.

[8] J. ZUPANCIC, S. WRYCZA, Information Systems Development. ISD '94, Proceedings The Fourth International Conference, University of Maribor, Slovenia, 1994.