

Using an Object-Oriented Methodology Called TAD in Business Process Reengineering

Talib Damij, University of Ljubljana, Slovenia

Abstract

The purpose of this paper is to introduce the use of an object-oriented methodology called TAD in the field of business process reengineering. TAD methodology consists of six phases and uses several tables to represent the real world of the enterprise. The first and second phases deal with identifying the reality of the system. The third phase focuses on carrying out business process reengineering in order to create a competitive and successful enterprise. The last three phases are used to develop the information system of the reengineered organization. A problem of Sales-Claim is used as an example to implement the concept of this methodology. As a result of this work we found that TAD methodology is very useful in identifying and implementing changes in the functioning and structural scheme of an enterprise.

1 INTRODUCTION

In recent years business process reengineering (BPR) became a very important way of ensuring changes in an organization's structure and functioning to create a better, more competitive and successful enterprise.

The aim of this work is to represent the use of an object-oriented methodology called Tabular Application Development (TAD) in the fields of Business Process Reengineering.

TAD represents a new concept, which is simple and very different from the ideas used in other methodologies. This paper does not compare this methodology with other methodologies, because such a comparison exceeds the scope of this work.

Using TAD methodology the functioning of the organization is shown using several tables. These tables are then analyzed with the purpose of identifying the necessary changes which have to be implemented to improve the functioning of the organization. Implementation of the suggested changes in the organization's operation is then put into effect by developing the information system of the enterprise.

TAD has six phases. These phases are problem definition, systems functioning, business process reengineering, object model, design and implementation.

A problem of Sales-Claim is presented to illustrate the use of TAD methodology in carrying out BPR.

2 PROBLEM DEFINITION

Problem definition is the first phase of TAD methodology. To carry out BPR in an enterprise we first have to identify its goals and objectives at strategic, business and operational levels and also to understand the structure of the enterprise.

To achieve that we start by organizing interviews with the top management. The purpose of these interviews is to identify:

- the organization's strategic plan and goals;
- vital analyses related to the strategic plan and goals;
- the organization's decision support problems; and
- the organization's structural scheme.

After identifying the strategic goals, we continue to organize interviews with management at business and then at operational levels. To do that a plan of interviews is created suitable to the hierarchical scheme of the organization.

The purpose of the interviews organized at business and operational levels is to identify:

- the business and operational objectives and goals;
- the structure of each entity (department or unit);
- analyses related to business and operational goals.

The result of the interviews is a list of strategic, business and operational goals and a list of analyses related to the listed goals. Each of these analyses shows the current achievement of a certain goal.

TAD methodology uses the term entity to define a user, group of users, unit, department, or any source of information that is part of the system or is connected with the system by some interaction.

Linking the identified analyses to the management at enterprise, business and operational levels is achieved by developing a table called the entity table. The entity table is structured as follows: the columns of the table represent the entities and the rows represent the analyses. An asterisk in any square (i,j) in the entity table means that the entity defined in column j is connected with the analysis defined in row i , where i ranges from 1 to the number of analyses and j ranges from 1 to the number of entities.

Sales-Claim: Sales-Claim represents a difficult problem in a large trading company. Many customers are not satisfied with the solution obtained and also to get a solution takes a long time. To resolve this problem the management decided to analyze it using



TAD methodology. During the first phase of TAD methodology interviews were conducted at different management levels and identified a strategic, business, and an operational goals. Some of these goals are:

- Minimizing the number of sales-claims,
- Renewing contracts with suppliers who purchase products with minimal claims, and
- Shortening the process and time of sales-claim solving.

Table 1 introduces only part of the real entity table. This table shows some important analyses related to the defined goals.

Requirements	Entity	Sales Management	Purchasing Management	Dispatch Management	Quality Coordinator
	Analysis				
Reports	1. Claims sorted by sales units	*	*		
	2. Claims sorted by customers	*	*		*
	3. Claims sorted by product groups	*	*		*
	4. Claims sorted by suppliers	*	*		*
	5. Information about claims causes	*	*	*	*
	6. Information about claims' solutions	*	*	*	
	7. Information about claim duration	*	*		

Table 1. Entity table of Sales-Claim

3 SYSTEMS FUNCTIONING

In order to carry out BPR we have to identify and understand the functioning of the enterprise. Therefore, an essential precondition for a successful BPR is a complete understanding of the real world of the system considered. This means that the analyst has to discover the functioning of the system as a whole and also the functioning of each department in the framework of the system.

This phase has three steps. The first step identifies the activities of the system. The second step describes the activities in detail. The third step defines the work processes and business processes.

Activities

The first step deals with identifying every activity performed in each part of the organization. Identification of the organization's activities is achieved by organizing further interviews with the representatives of every internal entity in the framework of the organization.

An effective way to identify the activities is achieved by developing an activity table. As we develop the activity table we simultaneously develop another table called the task table, which is discussed later.

The activity table is organized as follows: The entities are represented in the columns and the activities are listed in the rows of the table.

Every activity occupies one row. A non-empty square(i,j) shows a certain task performed by an entity defined in column j inside the activity defined in row i . A task is work performed by a determined entity in the framework of a certain activity.

Developing the activity table is a result of interviews organized with the internal entities defined in the columns of the table. In the rows of the activity table we first register each activity identified during an interview and then link this activity with the entities in the columns which cooperate in carrying it out.

To make the activity table represent the real world, we link the activities horizontally and vertically. The purpose of defining horizontal and vertical connections is to define the order in which they occur in the real world.

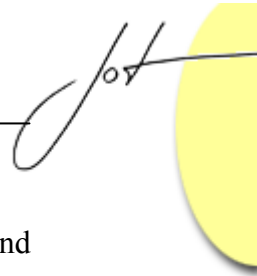
Horizontal linkage means that each activity must be connected with those entities in the columns, which are involved in performing tasks in the framework of a particular activity. To indicate that letters S and T are used.

Letter S in square(i,j) means that entity(j) is a source entity for activity(i). This entity(j) performs a determined task in the framework of activity(i) (creates, completes, sends etc.) defined in square(i,j).

Letter T in square(i,j) means that entity(j) is a target entity for activity(i). This entity(j) accepts or registers an output from other entities.

Usually, each activity is connected with two entities; these are the source and target entities. An activity could be related to only one entity. This occurs when an entity performs determined work for its purpose and which is not linked to any other entity. On the other hand, an activity could be connected with more than two entities. This is possible when an entity (entities) receives an input from another entity (entities) and sends it to one or more entities.

Therefore, any activity may have one or more source entities and also a number of target entities. For this reason the letters S and T, used to indicate a certain activity, are also indexed by the index of the source entities of the task as it occurs in the framework of the treated activity.



For example, considering activity(1) let us define that entity(3) is a source and entity(9) is a target entity. To do so we write S_3 in square(1,3) and T_3 in square(1,9).

Vertical linkage exists between activities performed by a certain internal entity. In other words, considering only the internal entities we use the letters P and U to connect the activities in which every internal entity is involved.

Letter P in square(i,j) means that activity(i) is a predecessor to some activity (activities) indicated by U in column j. Letter U in square(i,j) means that activity(i) is a successor to another activity (activities) indicated by P in column j.

Any activity may have one or more predecessors and also one or more successors. The letters P and U are indexed by the index of the predecessor activity.

For example, considering the entity defined in column 1 let us define that activity(2) is a predecessor to activity(4). To do this we write P_2 in square(2,1) and U_2 in square(4,1).

Furthermore, an activity may have more alternative successors. In this case, an OR operator is used to indicate each of the alternative successors. An OR operator is represented by an asterisk written on the right side of letter U.

For example, considering the entity in column 2, let us define that activity(1) is a predecessor to alternative activities 2 and 9. This means that from activity 1 we may continue either with activity(2) or activity(9). To define this we write P_1 in square(1,2) and U_{1*} in squares (2,2) and (9,2),).

Sales-Claim: The first and the second column of Table 2 are the subject of the third step. They are shown in Table 2 because of space limitations and should be ignored for now. To develop the activity table of the Sales-Claim, interviews were organized with the different entities defined in Table 2. This table shows a reduced activity table of the Sales-Claim, which has 18 activities and 6 entities. The first 5 entities are internal and the last one is external. The real activity table has 187 activities and 19 entities. The first activity "Receive Claim Note" means the Sales-Claim clerk receives a claim note from a customer. The second activity means that Sales-Claim clerk registers the claim note. The third activity means that Sales-Claim clerk prints a claim form. Thus to indicate the first activity we write S_6 in square(1,6), T_6 in square(1,1) and S_1 in square(2,1) and square(3,1) to indicate the second and third activity.

Furthermore, concerning the Sales-Claim clerk we find that the first activity is a predecessor to the second activity and third activity. For this reason we write P_1 in square(1,1), U_1 in square(2,1), and square(3,1). From the seventh activity we may continue either with activity 8, 15 or 16. To indicate this linkage we write P_7 in square(7,1) and U_{7*} in squares (8,1), (15,1) and (16,1). All activities are connected horizontally and vertically using the same concept.

Tasks

The activities registered in the previous step need to be defined in detail. To do that we describe their tasks and identify all the circumstances in which they are accomplished.

A very good way to identify tasks, their characteristics, and the circumstances linked with them is by developing a task table. As mentioned above the task table is developed simultaneously with the activity table.

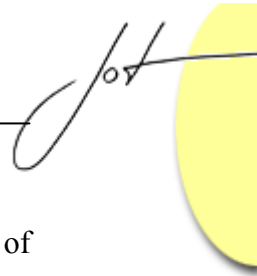
The task table is organized as follows: The tasks are represented by the rows of the table and the characteristics of the tasks are defined in the columns.

Business Process	Work Process	Entity	1.	2.	3.	4.	5.	6.	
		Activity	Sales Claim Clerk	Sales Clerk	Warehouse Cla. Clerk	Manager of Dispatch D	Stock Keeper	Customer	
Sales-Claim	1. Reception of Sales-Claim	1. Receive Claim-Note	T ₆ P ₁						S ₆
		2. Register Claim-Note	S ₁ U ₁						
		3. Print Claim-Form	S ₁ U ₁ , P ₃						
		4. Collect Claim Documentation	S ₁ U ₁ , P ₄						
		5. Send Claim Documentation	S ₁ U ₃ , U ₄	T ₁ P ₅					
		6. Determines Claim Solution		S ₂ U ₅ , P ₆					
		7. Return Claim Documentation	T ₂ P ₇	S ₂ U ₆					
	2. Under-Received Products	8. Forward Claim Documentation	S ₁ U ₇ *			T ₁ P ₈			
		9. Analyze Quantity of Products				S ₃ U ₈ , P ₉			
		10. Approve the Claim				S ₃ U ₉ , P ₁₀			
		11. Reject the Claim				S ₃ U ₉ , P ₁₁			
		12. Return Claim Documentation	T ₃ P ₁₂			S ₃ U ₁₀ , U ₁₁			
		13. Issue Credit-Note	S ₁ U ₁₂						T ₁
		14. Send Explanation	S ₁ U ₁₂						T ₁
	3. Over-Received Products	15. Send Additional Invoice	S ₁ U ₇ *						T ₁
	4. Warehouse-se Product Return	16. Inform Products Transport Back	S ₁ U ₇ *, P ₁₆				T ₁		
		17. Send Transport Schedule	S ₁ U ₁₆ , P ₁₇			T ₁ P ₁₇		T ₁ P ₁₇	
		18. Inform about Shipment Reception	T ₃ U ₁₇			T ₅ , S ₃ U ₁₇		S ₅ U ₁₇	

Table 2. Activity table of Sales-Claim

Thus each task, defined by a non-empty square in the activity table, occupies one row in the task table.

Each task is represented by its code K_{ij}, where the letter K means a task, and i and j are indices of row i and column j of the activity table where the task is defined.



In the columns of the task table we define task characteristics in terms of Description, Time, Condition, and Document.

Description is used to write a short description of the particular task.

Time is used to denote that entity(j) in the activity table needs a determined time to perform task K_{ij} . Time may become a very useful parameter should we wish to use it to reengineer business processes as discussed later.

Condition is used to define that performing task K_{ij} requires that one or more conditions be fulfilled.

Input/Output is used to indicate which inputs and outputs are connected with task K_{ij} .

In addition to the tasks, the task table represents the activities defined in the activity table. The purpose of this is to show in detail the linkage between the activities and their tasks.

Developing the activity and task tables is an iterative process. Some of the interviews have to be repeated to arrive at a precise understanding of the user's work. If anything is not understandable, then we have to organize a new interview with the responsible user until everything is clear.

Sales-Claim: Table 3 describes in detail only the first five activities of the activity table (Table 2). All other tasks are defined using the same procedure.

Activity	Characteristic	Description	Time	Condition	Input / Output
	Task code				
1. Receive Claim-Note	$K_{1,1}$	SCC receives a Claim-Note from the customer			Claim-Note
2. Register Claim-Note	$K_{2,1}$	SCC registers Claim-Note			Claim-Note
3. Print Claim- Note	$K_{3,1}$	SCC prints a Claim-form which enables the customer to open a claim		Check Customer's Claim-Note	Claim-Form
4. Collect Claim Doc- umentation	$K_{4,3}$	SCC collects the rest of the claim documentation			Dispatch- Order Invoice
5. Send Claim Documenta- tion	$K_{5,1}$	SCC sends claim documentation to SC			Claim Documenta- tion

Table 3. Task table of Sales-Claim

Work processes and business processes

In this step the analyst has to analyze the activity table to define the work processes of the system by grouping the activities into suitable collections. Each of these collections represents a determined work process. A work process is then a collection of activities. Therefore, each work process occupies one or more rows in the activity table.

The work process is the lowest-level group activity within the organization (Watson, 1994).

The analyst goes on grouping the work processes into suitable collections; each of these collections is a business process.

Business processes are groups of related work processes that deliver a common outcome (Watson, 1994).

Business processes are performed by combinations of work processes, while work processes are performed by individuals activity as a team (Watson, 1994).

After finishing this work we try to complete the entity, activity, and task tables. This is achieved by organizing a joint meeting with the main or important representatives of internal entities. At this meeting we present the entity, activity, and task tables to them. The aim of this presentation is to correct any mistakes or make necessary changes in the tables, and to persuade the representatives to complete the tables introduced.

Creating the activity table leads to the discovery of all activities, all work processes defined by grouping the activities in convenient collections, and all business processes identified by grouping the work processes in appropriate collections. Thus we can say that creating the activity table leads to the discovery of the complete functioning of the enterprise and its departments.

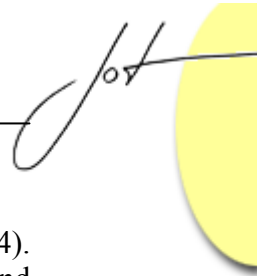
Sales-Claim: Corresponding to this step we grouped the activities defined in Table 2 into four work processes. These are: Reception of Sales-Claim, Under-Received Product, Over-Received Product, and Warehouse Product Return. These work processes are grouped into one business process; this is Sales-Claim.

4 BUSINESS PROCESS REENGINEERING

The third phase of TAD methodology deals with carrying out BPR in the enterprise concerned. More precisely, this phase deals with identifying and implementing changes to improve the functioning of the enterprise.

The third phase has two steps. The first step deals with partitioning the activity table. The second carries out BPR in the entity, activity and task tables.

BPR became a very important way of assuring changes in an organization's structure and functioning to create a better, more competitive and successful enterprise.



BPR is a method for identifying, defining and implementing change (Watson, 1994). This is achieved by defining an appropriate linkage between the strategic, business and operational levels of an organization.

At the enterprise level of the organization, top management focuses on the strategic plan and goals by developing organizational values, vision, mission and objectives. The enterprise level is concerned with the implementation of the strategic plan and goals at business and operational levels.

At the business level of the organization, the strategic plan and goals are deployed within the context of its market environment and translated into business objectives and goals. This level tries to meet customer needs and expectations, and develop appropriate relationships with customers, suppliers and business partners.

At the operational level of the organization, the business objectives and goals are translated into action plans. The principal work of the operational level is to maintain reliably the flow of both products and services to customers.

The developed entity, activity and task tables generate a clear and visible linkage between the enterprise, business, and operational levels of the organization.

The enterprise level is introduced by the analyses related to the strategic plan and defined in the entity table.

The business level is represented by the analyses connected to business goals and objectives and defined in the entity table. In addition to this, the business level is introduced by a set of business processes defined in the activity table.

The operational level is introduced by a set of analyses related to operational goals and defined in the entity table and also by a number of work processes and activities defined in the activity table and described in the task table.

Table partitioning

Having the entity, activity and task tables before him enables the analyst to understand the functioning of the enterprise and offers him an opportunity to participate in creating a better and more effective enterprise. This goal is achieved by precise analysis of the entity and activity tables, suggesting changes and improvements, and giving solutions for existing problems.

To do this, we first have to establish a project team. This team consists of the analyst and representatives of the enterprise, business and operational levels.

We start by analyzing the activity table because this table represents the existing functioning of the system. The purpose of such an analysis is to create a complete understanding of the activity table, which is an essential precondition to move forward in carrying out BPR.

The activity table has to be analyzed carefully. This is very important because the activity table developed in the previous phase may have one or more business processes and each business process could contain many work processes.

In fact the activity table is already divided into several parts, each of them representing a certain business process. The team continues by looking for the possibility to divide each business process, particularly those with many work processes, into several parts without losing information or connections existing between work processes.

This process is called table partitioning. The aim of table partitioning is to facilitate the project team's work. The concept of table partitioning is based on the idea of identifying basic work processes in the framework of each business process.

A basic work process is a work process related to different alternative work processes for performing alternative jobs. None of these jobs could be done without the basic work process. A non-basic work process is recognized in the activity table by an OR operator indicated inside a certain task of its first activity.

Table partitioning is therefore done by dividing a certain business process into parts starting with a basic work process and including an alternative path of a set of non-basic work processes. So, each part of the table contains, in addition to the basic work process, a set of one or more non-basic work processes.

Sales-Claim: Table 2 has one basic work process and three non-basic work processes because from the last activity of the first work process we may continue with the second, third or fourth work process. This is indicated by an asterisk (OR operator) in squares (8,1), (15,1) and (16,1). Therefore, Table 2 could be partitioned into three parts. The first part contains the first and the second work processes. The second part contains the first and the third work processes. And the third part contains the first and the fourth work processes.

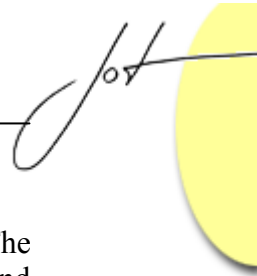
Table reengineering

The result of table partitioning is creating several parts (subtables) of the activity table. Each of these parts could be analyzed separately. Therefore, the team analyzes and discusses each part carefully in order to understand it completely. As a result of such an analysis the team may find ideas to improve the functioning in each part discussed by addressing the possibility of:

- shortening work processes;
- removing redundant activities or tasks;
- moving tasks from one entity to another, if these tasks could be accomplished better;
- shortening the times needed to perform time-consuming activities or tasks;

The task table contains valuable information about the time needed to perform any task. Therefore, the team can quickly find the most time-consuming tasks and try to find better solution.

After creating a complete understanding of each part, analyzing it carefully, and trying to improve it, we continue with building a unified activity table again by connecting the parts together.



The improved activity table enables us to start with BPR at a higher level. The project team analyzes the existing strategic plan and goals, makes the needed changes and defines a new strategic plan and goals if necessary. They also update the entity table in accordance with the suggested strategic plan.

The team continues its work by considering the business processes defined in the activity table and creating a linkage between the enterprise level and the business level. This is achieved by making any necessary changes in the existing business processes or defining new business processes corresponding to the new strategic plan.

The team goes on analyzing the work processes defined in the activity table and then defining an appropriate linkage between business and operational levels. This is achieved by making changes in the existing work processes in accordance with changes made in the business processes. These changes are also made in the activities listed in the activity table. In accordance with these changes the task table is also updated. The aim of these changes is to optimize the time and resources needed to perform any work process.

The project team's work leads to reengineering the entity, activity and task tables which is essential in creating a more successful organization. This goal is achieved by:

- defining new strategic goals, making the needed changes in the existing ones, or removing obsolete goals;
- creating an appropriate linkage between the strategic goals and business processes by defining new business processes, changing the existing ones, or removing the unnecessary business processes;
- creating a convenient linkage between the business processes and work processes by translating the changes made at the business level into the operational level of the organization.

In addition to making changes in the strategic plan and goals, business processes and work processes, the project team has to analyze the obligations of the internal entities defined in the columns of the activity table and to find a solution to the existing problems about:

- merging entities together;
- closing unneeded entities;
- opening new entities (departments).

TAD methodology makes the achievement of BPR possible using the information collected in the entity, activity and task tables. These tables enable the team to analyze the strategic goals, business processes, work processes, activities and tasks from various points of view.

Sales-Claim: Corresponding to the current step Tables 1, 2 and 3 were analyzed carefully. We found that the claim documentation is in permanent movement between three different entities which are involved in performing similar activities. These entities are: sales-claim clerk, warehouse-claim clerk and purchase-claim clerk (this clerk is not shown in Table 2 but is in the original one). For this reason, we suggested unifying these jobs by defining one entity; this is claim clerk, who takes care of the claim. In addition to

this, we suggested shortening the duration of some work processes (these work processes are not shown in Table 2).

5 OBJECT MODEL

To implement the ideas of changes in the functioning of the enterprise identified in the third phase we continue with the information system development. The fourth phase deals with developing the object model of the system. This phase has two steps. Before describing these steps let us introduce some important definitions:

An object is anything, real or abstract, about which we store data and those operations that manipulate the data (Martin, 1993).

An object has state, behavior, and identity; the structure and behavior of similar objects are defined in their common class; the terms instance and object are interchangeable (Booch, 1994).

An object is characterized by a number of operations and a state which remembers the effect of these operations (Jacobson, 1996)

An object is simply something that makes sense in an application context (Rumbaugh et al., 1991).

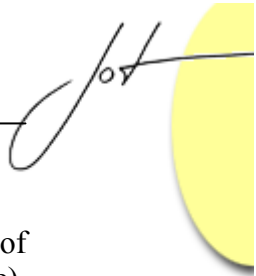
In the first step we identify the object classes, their attributes and their associations. Information about object classes is obtained from interviews with the users and by analyzing the entity, activity and task tables. The classes, attributes and associations identified are then used to develop an initial object model.

We start by analyzing the task table. For this purpose, we carefully consider each description defined in the Description column to identify object classes, attributes and associations which are mentioned in it. The result of this analysis is a group of object classes, their attributes and their associations. We continue by analyzing the other columns of the task table, especially the inputs and outputs defined in the Input/Output column. If any new object class, attribute or association is identified, the existing group of classes is extended by it.

This work is continued by analyzing the activity and entity tables, particularly the outputs and reports defined in the rows of the entity table. Any newly discovered object class, attribute or association is added to the existing ones. In addition to this, the analyst may add any other object class if it is useful to the application.

Such an analysis is carried out using three types of relationships that can exist between classes and attributes. These relationships are the Is-part-of, Is-associated-with, and Isa structures.

A class of objects can be constructed, described and qualified by the objects' attributes and by other classes (the Is-part-of structure). A class, then, is constructed from components consisting of attributes and other classes. This is referred to as the class aggregation abstraction (Sanders, 1995).

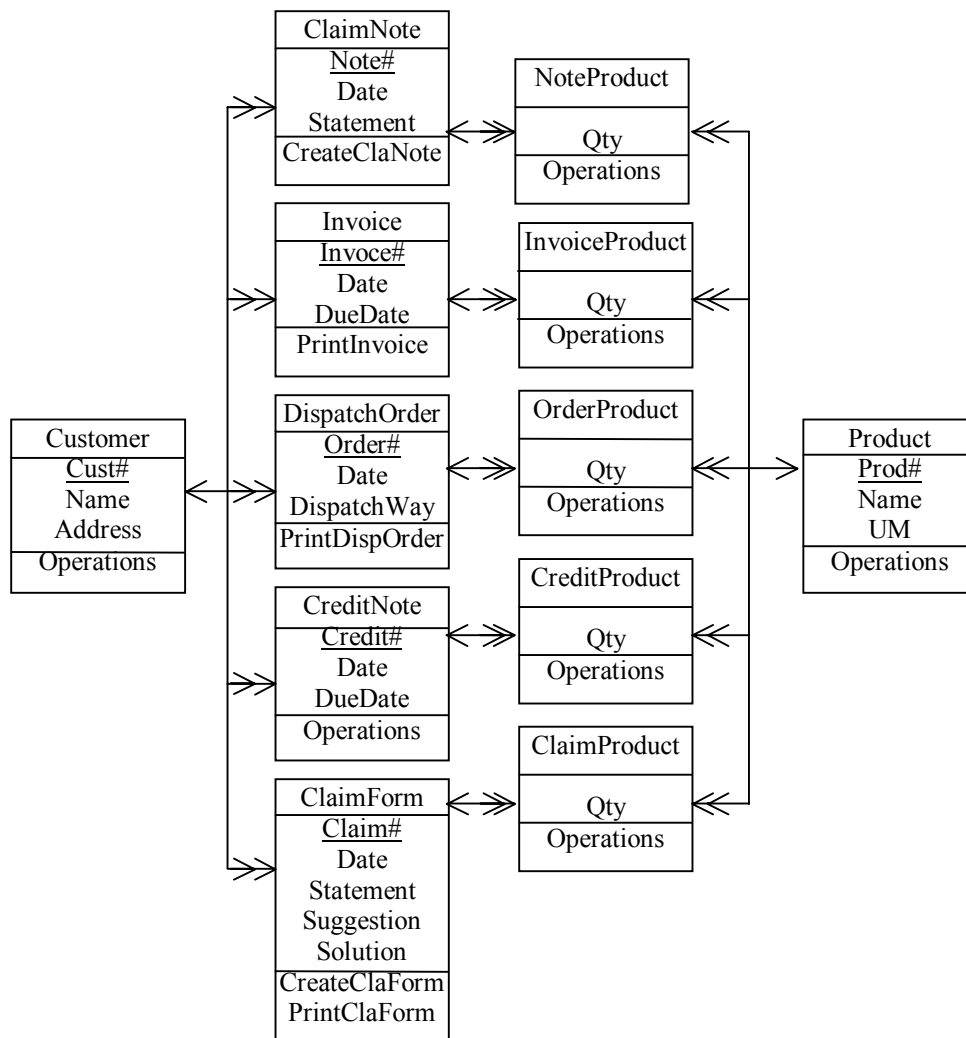


A class can interact with other classes, and a class can affect the attribute values of other classes through associations or collaborations (the Is-associated-with structure). This is referred to as the class association abstraction (Sanders, 1995).

In the first step we try to develop the initial object model by implementing the Is-part-of and Is-associated-with structures.

A class can be related to other classes through subclasses (the Isa structure). Thus a subclass can inherit attributes from superclass, and it can have its own unique attributes. This is referred to as the class generalization and specialization abstraction (Sanders, 1995).

In the second step we try to organize the classes of the object model obtained in the previous step using inheritance (the Isa structure) to share common structures.



Sales-Claim: In accordance with the first step of the current phase, we carefully analyzed the task, activity and entity tables to create the initial object model. This object model is simplified in comparison with the real initial object model.

Corresponding to the second step of this phase we implemented the inheritance between the classes of the initial object model. Because of space limitations Figure 1 shows only the final object model. For the same reason, this model also shows some operations in the framework of object classes. Identification and addition of operations to the object model is discussed in the first step of the next phase.

6 DESIGN

The fifth phase of TAD methodology deals with designing the system and preparing it for implementation. This phase has two steps. The first step specifies the operations of the object model. The second step develops the application model of the system.

Operations

Identification of the operations requires creating a linkage between the tasks defined in the task table and the classes of the object model.

This is achieved by analyzing the tasks described in the task table and trying to link each task with the classes involved in performing it. If any connection exists between the task considered and any class of the object model, then an operation is defined in the framework of the class.

Usually each task is transformed into an operation. In this case, the task is simple and deals with one or more successive

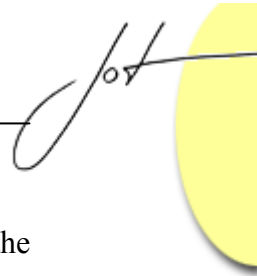
events. Sometimes more than one operation are identified by analyzing the task description in the task table.

To register the linkage created between each task, operations identified from it, and classes to which the operations belong we develop a new table called the operation table.

The operation table consists of three columns. The first column lists the tasks as they are defined in the task table. The second column represents the operations identified by analyzing tasks listed in the first column. In the third column we register the classes which contain the listed operations.

Each operation occupies one row in the operation table. Any task may be connected with one or more operations. For this reason, each task occupies one or more rows of the operation table, depending on the number of operations related to the task considered.

After registering each operation in the operation table and into the classes of the object model, we continue our work by defining each operation in detail. To do this we write an algorithm for each of the specified operations.



Sales-Claim: Table 4 shows the operation table of Sales-Claim. Figure 1 shows the object model extended by the names of the operations defined in the operation table.

Task code	Operation	Class
K _{1,1}	CreateClaNote	ClaimNote
K _{2,1}	CreateClaNote	ClaimNote
K _{3,1}	CreateClaForm	ClaimForm
	PrintClaForm	ClaimForm
K _{4,3}	PrintDispOrder	DispatchOrder
	PrintInvoice	Invoice

Table 4. Operation table of Sales-Claim

Application model

Development of the application model is derived from the information collected in the entity, activity and task tables. The application model consists of several parts.

The first part is derived from the entity table and represents the vital analyses defined in this table. This part is especially important for management, because they deal with various analyses that are essential to decision making because they are related to the strategic plan, enterprise, business and operational objectives and goals.

The other parts of the application model are developed in accordance with the information stored in the activity table.

These parts are derived from the first column of the activity table. Each of them represents a particular business process. Every business process may have one or more work processes corresponding to the content of the second column of the activity table. Every work process includes a number of activities in accordance with the content of the third column of the activity table.

After completing the application model our work is continued by writing a set of algorithms which define the created application model in detail.

Sales-Claim: Figure 2 shows the application model of Sales-Claim. This model consists of two parts. The first part is created corresponding to the content of the entity table. The second part is developed corresponding to the content of the activity table.

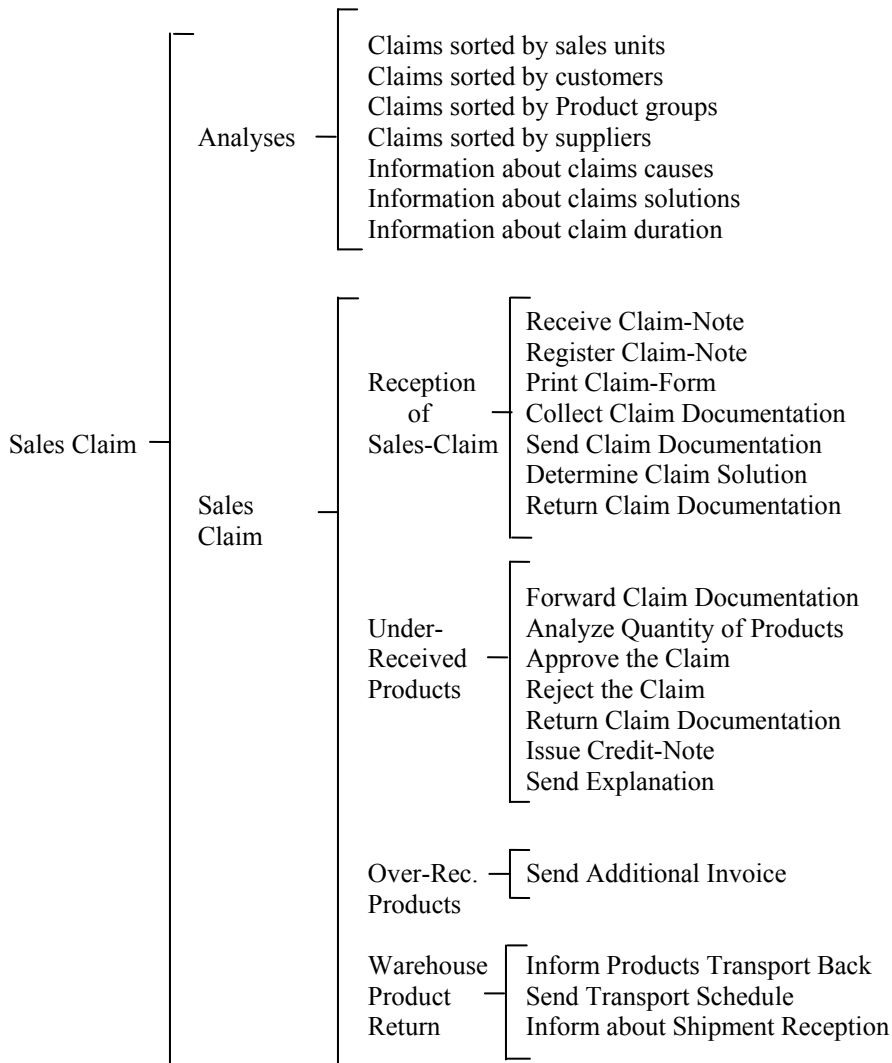


Figure 2. Application model of sales Claim

7 IMPLEMENTATION

The sixth phase of TAD methodology deals with the implementation of the system developed in the previous phases. The inputs to the implementation phase are the object model, the application model, and the algorithms related to both models.

In the implementation phase we deal with choosing a convenient data base management system to implement the object model of the system and translating the written algorithms related to the operations of the object and application model into program codes.



8 CONCLUSION

The aim of this work was to find out if TAD methodology could be used in the field of BPR as an approach to identifying and implementing changes necessary to improve the functioning of the enterprise. We found that using TAD methodology leads the analyst to achieve this goal in an easy manner. This is because using tables to represent the real world of the enterprise is effective, understandable, visible and easy. Tables can be analyzed in order to define linkages between strategic, business and operational levels of the enterprise. Tables can also be analyzed horizontally to identify the possibility of merging entities which perform similar jobs, to define new entities, or to remove unneeded ones.

Using TAD methodology in the field of BPR enables systems analysts to play a major role in enterprise reengineering contributing to making a successful and competitive enterprise.

REFERENCES

- [Booch94] Grady Booch: *Object-Oriented Analysis and Design with Applications*, Benjamin-Cummings, California, 1994.
- [Booch94] Grady Booch: *Object Solutions, Managing the Object-Oriented Project*, Benjamin-Cummings, California, 1994.
- [Hammer, Champy93] Micheal Hammer and James Champy: *Reengineering the Corporation, A Manifesto for Business Revolution*, HarperBusiness, New York, 1993.
- [Jacobson96] Ivar Jacobson: *Object-Oriented Software Engineering, A Use Case Driven Approach*, Addison-Wesley, Wokingham, 1996.
- [Jacobson95] Ivar Jacobson: *The Object Advantage, Business Process Reengineering with Object Technology*, Addison-Wesley, Wokingham, 1995.
- [Martin93] James Martin: *Principles of Object-Oriented Analysis and Design*, Prentice-Hall, Englewood Cliffs, 1993.
- [Rumbaugh91] James Rumbaugh: *Object-Oriented Modeling and Design*, Prentice-Hall, Englewood Cliffs, 1991.
- [Sanders95] Lawrence Sanders: *Data Modeling*, Boyd & Frase, Danvers, 1995.
- [Watson94] Gregory Watson: *Business Systems Engineering: Managing Breakthrough Changes for Productivity and Profit*, John Wiley & Sons, New York, 1994.

About the author

Talib Damij received his PhD (Information-Management Science) from the faculty of Economics at University of Ljubljana in 1988. He is currently as associate professor at the faculty of Economics, University of Ljubljana. Dr. Damij's research interests include information systems development methodologies, information systems project management and business process reengineering.