Company strategy (and objectives) is managed by activities. Success of strategies is directly related to the efficiency and effectiveness of activities. Activities are like road maps to reach the destination (objectives).

Activity Modeling and Business Process Re-engineering

Company strategy covers 4 broad areas:

1. Market place chosen to be served
2. Identifying Services and products for this market
3. Marketing and Sales channels to access this market
4. Activities and Processes to apply Strategies

Activity (or Process Models)

Activities are hearts of organizations.

Activity Model: Defines the organization as a system of interrelated activities

Activity (or Process Models)

Activities are difficult to express as texts.

Which one is easier to understand?
Activity (or Process Models)

* Activities are difficult to express as texts.

Which one is easier to understand?

Activity Model vs Process Model

- In activity models, the job (business function) is more important than the actor who performs the job.
- In process models (or process flow models), the actor who performs the activity is also important (e.g. Process models in Oracle Designer).
- BPwin supports both methods

Why Modeling?

In all types of organizations, the quality and success result from the processes, or activities.

Why Modeling?

Modeling is essential to understand the interactions between the activities and communicate to make them more effective and efficient.

Using CASE tools, you can have automatic support for Information System Development.

Why Modeling?

Helps the software developer in Form Design

Why Modeling?

Integration with Data Modeling
What is Process Modeling?

* An effective method to understand business rules and processes
* You can hide the details and focus on required parts
* Visual Objects (using graphics) facilitate communication
* A standard for Quality Management

History

* SADT (Structured Analysis and Design Technique) was developed by SofTech in 1960. It is the basis for IDEF0 notations.
* US Airforce accepted it as a standard in 1970s and DoD (Department of Defence) adopted as a standard
* In 1993, IDEF Users Group (Society of Enterprise Engineering) worked together with NIST and produced FIPS (Federal Information Processing Standards)
* IDEF3 was developed by Air Forces and DFD was developed independently.

Process Modeling using BPWin

Using BPWin;
You can understand, analyze, and document complicated processes.

Benefits

* Removing the activities which are repetitive and insufficient
* Reducing costs
* Increasing flexibility
* Increasing Customer relations
Business activity diagrams

Inputs, Controls, Outputs and Mechanisms

Picture of all processes within the organization

Necessary for Form Design

BPwin: Types of Process Modeling

BPwin supports 3 modeling methods.

- Business Process Modeling (IDEF0)
- Process Flow Modeling (IDEF3)
- Data Flow Modeling (DFD)

Business Process Modeling (IDEF0)

- IDEF0 (Integration Definition Language) is activity modeling technique
- All system is described as set of interrelated activities and functions.
- System activities (actions) are examined independent of the objects which perform these activities

Example: A simple Business Process Diagram

Data Flow Diagram (DFD)

- Data Flow Diagrams are used to model data flow between data stores and activities

Example: Data Flow Diagram

Data Flow Diagram (DFD)

Arrows in data flow diagrams show the movement of data between activities.

4 simple elements
- Arrows
- Data Store
- Activity (Process)
- External Reference (or entity) (Source/sink)

Example: Data Flow Diagram

Process Flow Modeling (IDEF3)

It is also called Process Flow Modeling.

Main Objective: To model activities (processes) in forms of sequential events.

Example: Data Flow Diagram
Process Flow Modeling (IDEF3)

IDEF3 diagrams are similar to Business Process Models in Oracle Designer. Example: Data Flow Diagram

Three Models in SDL:

IDEF0: Suitable for Analysis and Logical Design. Can also be used for all modeling.
IDEF3: Can be connected to IDEF0 diagrams
DFD: For Design purpose

BPwin Interface

BPwin Context (Root) Diagram

Business Process Models
System is comprised of interrelated activities.
The top activity is called root (context) activity. Root activity defines the system. The lower diagrams in hierarchy called decomposition diagrams.
BPwin Context (Root) Diagram

A root activity for IDEFO

ROOT ACTIVITY DIAGRAM
Each activity name is combination of a verb and an object name describing the activity objective.

Use of Arrows

In IDEFO diagrams, different arrow types are connected to activities. Arrows represent people, places, concepts or events. Arrows are labeled with different names and connected between activities and borders.

Use of Arrows

Input Arrows
Inputs represent material or information which are consumed or transformed to produce outputs.
Inputs are connected from the left hand side of activities.

Control Arrows
Controls impose rules that regulate how, when, and if an activity is performed and which outputs are produced.

Control Arrows
While they affect the activity, they are never consumed or transformed.

Activity Name: Verb+ObjectName

GET ORDER

ICOM (Input-Control-Output-Mechanism)

C

M

O

I

BORDERS

BORDERS

Product Information
Customer Arrow
Market Search Data

Ordering Procedure

GET ORDER

readonly
Use of Arrows

Control Arrows
Controls can also be used to describe items that trigger an activity to start or finish. For example, the receipt of a Customer Order will trigger the Order Confirmation process.

Mechanism Arrows
Mechanisms are physical resources that perform the activity. Mechanisms could be the important people, machinery, and/or equipment that provide and channel the energy needed to perform the activity.

Output Arrows
Outputs are the material or information produced by the activity. An activity that does not produce a definable output should not be modeled (or, at a minimum, should be a candidate for elimination). Output arrows always exit from the right side of an IDEF0 activity box. Each activity should have at least one output arrow. The example above illustrates two different outputs produced by the Get Order activity.

Decomposition Diagrams
Decompositions are used in business process modeling to break an activity into its constituent activities. Each of these activities can in turn be decomposed into its own constituent activities. Each time you decompose an activity, you create a decomposition diagram. The number of decomposition levels is entirely up to you, and depends on the level of complexity you need to model.

Note the yellow bubble on the figure. When an activity has not been decomposed, the "leaf" symbol will appear in the upper left corner of the activity box (called a "leaf-level" activity). After decomposition the leaf symbol is removed.
Decomposition Diagrams

Decomposing Activities into Components

1. First choose the activity to be decomposed and then choose child tool from the tool box.
2. Choose the method of decomposition diagram from the Activity Box Count dialog box.

Decomposed activity becomes main activity. Arrows are inherited to the decomposition diagrams. Arrows in decomposition diagrams are then connected to suitable activities.

5 different usages are available.
1. Output-Input
2. Output-Control
3. Output-Mechanism
4. Output-Control Feedback
5. Output-Input Feedback

Use of Arrows

1. Output-Input
After Customer Credit Information is Approved, it is sent to Approve Order Activity to be approved.

2. Output-Control
In this type, the first activity has an affecting (control) role on the second activity. It controls the second activity on how to convert inputs (accepted plans) into outputs. In the following example, the accepted plan is not changed and it controls the implementation of plan.
Use of Arrows

3. Output-Mechanism

This connection is very rare. It shows that output of an activity is a resource or method to perform another activity.

Following example illustrates that the fixture chosen after Choose Fixture activity is used as mechanism when machining the part.

4. Output-Control Feedback

Output is used as feedback control. After evaluating the project performance the performance result is used as control in developing a new project plan.

5. Output-Input Feedback

Feedbacks of this kind are frequently used in recycling processes. In the following example, after controlling part quality the part to be re-painted is input to be cleaned and re-painted.

Splitting and Merging Arrows

Outputs of an activity may be used by one or more activities. Policy and Procedures obtained from activity 1 are used in activities 1 and 2.
Splitting and Merging Arrows
Outputs of activities 1 and 2 are called Defected Materials.

Call Arrows
Call arrows are used to refer to the other models or diagrams inside a model.

Tunnels
Tunnels are used to pass between decomposition and parent diagrams.

Resolving Tunnels
They are resolved in 4 ways:
- Resolve as border arrow
- Resolve as rounded tunnel
- Resolve as Off Page Reference
- Resolve as External Reference

Other IDEF0 Diagrams
FEO (For Exposion Only) Diagram
It is produced by copying all objects in main diagram.
It is used to produce different scenarios and evaluate different point of views.
Other IDEF0 Diagrams

FEO (For Exposition Only) Diagram

It can be added by Diagram/Add FEO Diagram.

You can go to ‘Sibling’ FEO diagram.

Splitting and Merging Diagrams

Activities to be modeled inside projects can be split into smaller parts according to project groups and then merged later.

Splitting Model

1. Choose the activity to be split. This activity will be root activity.
2. Enter Split Model choice
3. Give the same activity name and then split it by using Copy entire dictionaries choice.

Models built separately can later be merged.
Merging Two Models

Use the following steps to merge two models together:

1. Open both the source and target models.
2. Right-click on the “placeholder” activity or the Call arrow in the target model where you want to insert the source model.
3. Select Merge Model on the shortcut menu.
4. Select the appropriate merge options in the Continue with merge? dialog.
5. Click OK to finish.

Adding Value to Activities

BPwin provides a framework of features that you can use to add value to your model. You can specify characteristics such as cost, time, and other properties you define.

There are two methods to specify such information:

* Activity-Based Costing (ABC)
* User Defined Properties (UDPs)

Activity-based costing (ABC) is a technique for capturing and analyzing activity costs to determine the cost of the overall process. BPwin supports a simple implementation of ABC where you can assign costs to activities in terms of currency and time.

For example, you can use ABC to determine:

- The cost of production.
- The cost to support customers.
- The cost of proposed process changes.
- Where costs are accruing in your business processes.

To calculate model costs using ABC, you must follow these basic steps:

1. Define the ABC units of measurement.
2. Define the ABC cost centers.
3. Apply costing estimates to all activities in the model.

Defining the Units of Measurement

First, you must define the currency and time units to use in calculating costs. Currency units are usually measured in U.S. Dollars, though you can choose other forms of currency. Time units can be measured in seconds, minutes, hours, days, weeks, months, or years.

These units are global for the model and are defined in the ABC Units tab of the Model Properties dialog. The units you select display on activities and reports.
Next you must create the ABC cost centers. Cost centers are categories of costs which are shared across all activities.

You can specify cost centers in the Cost Center Editor which can be opened from the Model menu.

After you define ABC units and cost centers, you can specify the cost of performing that activity for each cost center.

To assign activity costs, double-click the activity to open the Activity Properties dialog, and click the Costs tab. Then specify:

- The cost per cost center.
- How often the activity occurs (Frequency). This is usually set to 1.
- How long the activity takes to perform (Duration).

Costs for each process are calculated from the subordinate activities automatically.

Objects such as Lists, Sources, Windows, Commands can be defined in UDPs. They can be defined using UDP Definition Editor and called from related activities.

Activity containing UDP

Swim Lane Diagrams can be constructed Using UDP Definitions

Defined entities and attributes can be exported to ERwin.
Entity-Attribute Definition inside BPwin

Defining Data Usages

CRUD and IRUN Definitions

CRUD (for Entity)
C-Create
R-Retrieve
U-Update
D-Delete

IRUN (for Attribute)
I-Insert
R-Retrieve
U-Update
N-Nullify

DFD RULES

1. No process can have only outputs

2. No process can have only inputs

3. Data cannot move directly from one data store to another

4. Data cannot move directly from an external entity to a data store

5. Data cannot move directly from a data store to an external entity (sink)
Data cannot move directly from an external source to an external sink.

A data flow has only one direction of flow between symbols.

A fork in data flow means only the exactly same data goes from a common location to two or more different processes.

A join in a data flow means that exactly same data goes from any of two or more different processes, data stores to a common location.

No recursive data flow on a process.

A data flow to a data store means update (delete or change).
A data flow from a data store means retrieve or use.