COMMUNICATING CONCEPTS IN AN INTERDISCIPLINARY PROJECT

FOUR MODELS OF A LAKE DESCRIBED IN THE DELTA LANGUAGE

by
Birger Møller-Pedersen

DAIMI PB-77
May 1977

Institute of Mathematics University of Aarhus
DEPARTMENT OF COMPUTER SCIENCE
Ny Munkegade - 8000 Aarhus C - Denmark
Phone 06-12 83 55

Preface

This report is the first part of an English version of a Master's Thesis.

In this Thesis the DELTA language was used in the description of four different models of the same system, the experiences with the use was reported and an extension of DELTA by a context concept was proposed.

This report covers the use of the DELTA language. A forthcoming report will deal with the proposed extension by a context concept.

This report is also referred to as DELTA Project Report No. 7.

Århus, May 1977

Birger Møller Pedersen

Contents

INTRODUCTION

The work presented here can be seen as the following up of the work with system description by use of the DELTA language in DELTA project report no. 3 (DAIMI publ. no. 42, DELTA 74). The purposes have been

- to use the DELTA language for description of a more comprehensive system than the system considered in DELTA 74

- to evaluate the use of DELTA in this field

  and

- to propose a definition of and the properties of a context concept in DELTA.

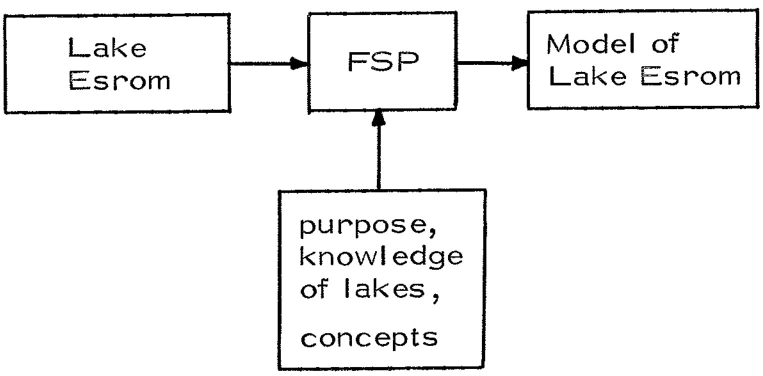
The work with system description in DELTA 74 was the first use of the DELTA language and the considered system should be described as seen from only one point of view. The primary interest was therefore to use the language as it was and as it developed through the use of it.

The system considered here is an eutrofication model of a lake. The model has been worked out by people from different disciplines, organized in "Fælles Sø Projekt" (FSP, "Common Lake Project"). The purpose of the co-operation in FSP has been to achieve new knowledge about lakes and their reaction upon changes due to external actions. To this end there has been worked out a lake model - built on investigations of Lake Esrom.

The lake model has thus been made by considering the same system from more than one point of view, by using concepts from different disciplines. In such an inter-disciplinary co-operation on working out a model it is essential that the models of the single disciplines are communicated the best to the people from the other disciplines and especially to the group having the responsibility of making the joint model. Would this communication be improved and easier to establish if a common system description language, e.g. the DELTA language, were used? I have tried to answer this question by – together with members of FSP – making descriptions in DELTA of the single models (of the different disciplines) and by communicating these to other FSP members. These descriptions and an evaluation of the use of DELTA are found in the chapters 1 and 2 of this report.

In FSP Lake Esrom has been considered to make a model of it:

The model of Lake Esrom has not been made only on basis of investigations of the lake. The purpose of FSP determines which of the aspects of the lake are shown in the model and determines how detailled the model is. The knowledge of other lakes and of lakes in general given in FSP influence the model, and the model is worked out using concepts from different disciplines. We say that the model is made in a certain context:



The context concept and its introduction into DELTA is the subject of a later report. In this report experiences with the use of a context concept and a formulation of the problems to be solved in the forthcoming report are presented.

Chapter 1

Use of the DELTA Language to describe different Models of a Lake

As mentioned in the Introduction the goal in FSP has been to make a model of a lake. A lake is an ecosystem so complex that it has been necessary to let a number of groups, each covering a discipline, treat different aspects of the system.

The work in FSP started with a discussion of and a clarification of the components of a lake model and of the mechanisms describing the dynamics of the system. These mechanisms formed the basic of the establishment of a number of groups, each treating one of the aspects: hydrology/hydraulics, grazing (eating of algae), growth of algae, the cycles of nutritive salts, topography/morphography and metals.Further a model group was formed; the task of this has been to composite the various submodels, working out a mathematical model and on this basis a DYNAMO model (see fig. 1.1, p.4).

The studying of FSP reports and relevant literature and talks with members of the FSP group have formed the basis of my description in DELTA of parts of the lake model. As the work in FSP, at the time when I started, was in such an advanced stage that the work with the DELTA system description could not be expected to influence the FSP model considerably, I have described an alternative model, including aspects not given in the FSP model.

I have confined myself to describe parts of the following submodels:

- topography/morphography
- hydrology/hydraulics
- biology

Further I have treated the mathematical model and the DYNAMO model.

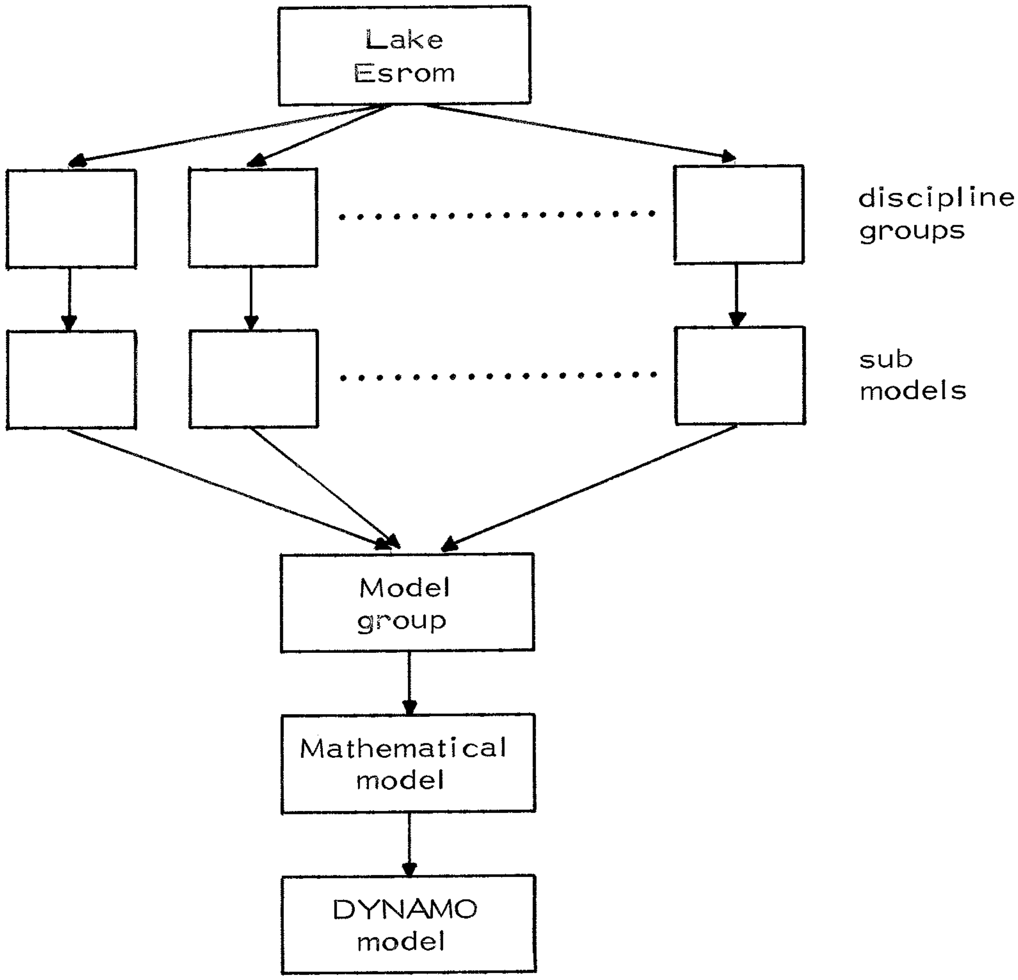


Fig. 1.1 Stages in the work of FSP

The descriptions have been worked out together with and have been directed to members of FSP. As they had little or no knowledge of DELTA the necessary concepts are introduced. Further a description of basic lake concepts is included.

The descriptions are examples of descriptions of models, so they appear well-balanced and finished. The considerations done during the work with the descriptions are reported in comments, inserted in the descriptions. They have not been part of the material of communication with FSP members. A survey of the experiences with the use of DELTA in this context is given in chapter 2.

The present DELTA language differs on minor elements from the version used here. The differences will be mentioned when first encountered.

1.1 Systems, Models and System Descriptions

The DELTA language is a language for description of DELTA systems, which are considered suitable as model systems for given systems.

By system is here meant a part of the world

- which we choose to regard as a whole, separated from the rest of the world

  and

- a whole which we choose to consider as containing a collection of components.

A model system (or model) is a system depicting certain and by us chosen aspects of a system.

In FSP Lake Esrom has been considered as a system and the result of the investigations has been a (mental) model of the lake, a model existing in the minds of the members of FSP.

In order to communicate this model it must be described in a language; we say that a model system description is made. The considered system is in this communication situation called the referent system.

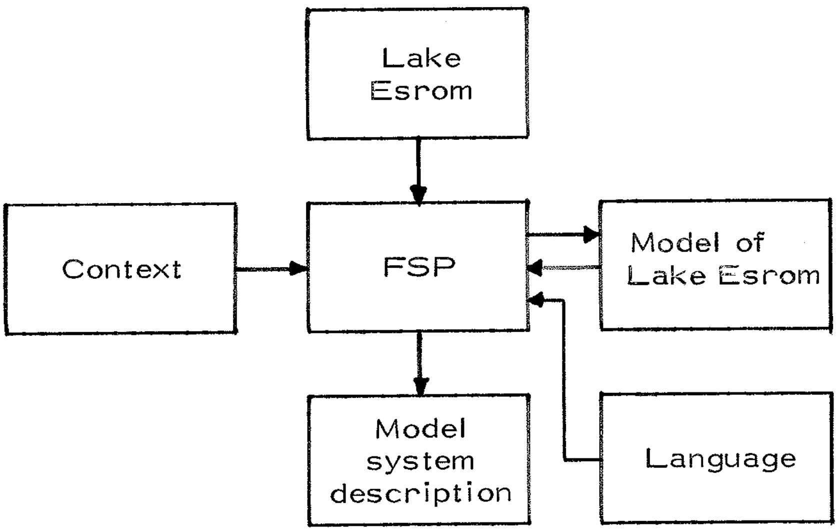


Fig. 1.2 Referent system, model system and system description

Besides for use in communication the working out of a system description has the purposes:

- to provide a tool for maintaining the mental model and for the working out in details of the model.

- to make the transformation from the mental model to a operational model easier.

FSP has worked out a model system description in a combination of danish and figures, tables and curves. An operationel model in the simulation language DYNAMO is based upon a mathematical model grounded on the concepts of Industrial Dynamics (System Dynamics, see GORDON 69, PUGH III 63).

When the DELTA Language is used as system description language we talk about DELTA (model) systems and DELTA system descriptions.

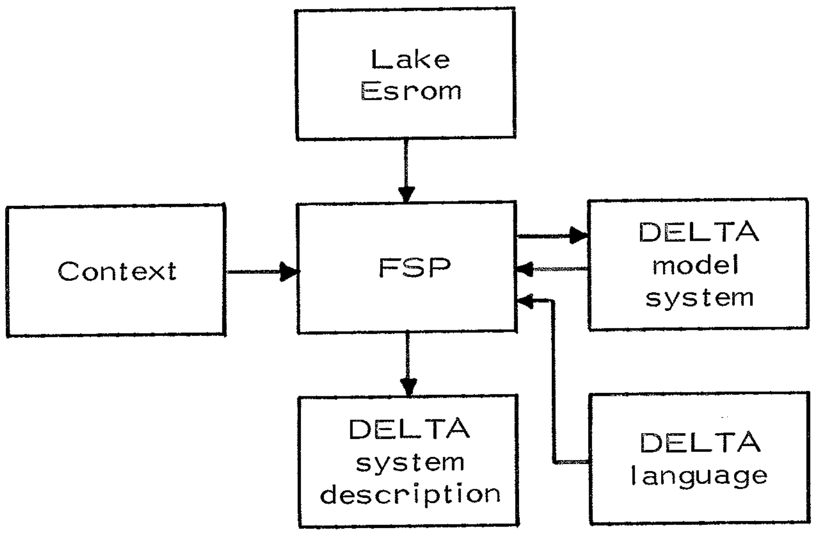


Fig. 1.3 The DELTA Language as system description language

By using knowledge of the system description language a receiver of the system description may generate a model of Lake Esrom, by interpreting the description. Also the system generator use a context when generating his model system.

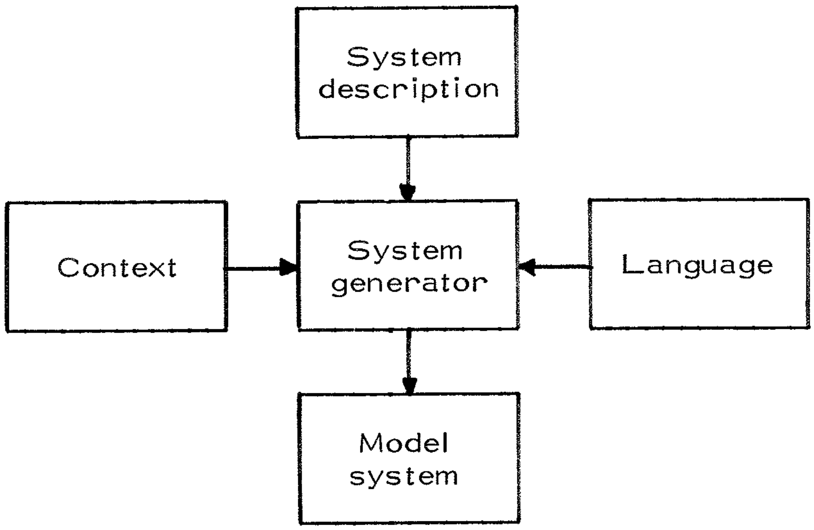


Fig. 1.4 The system generator generating a model system

The two model systems (the model of FSP and the model of the system generator) are different models of the referent system. They are either generated by different generators or they are generated at different points of time. It is essential to obtain that the model system of the system generator is as similar as possible to the model system of FSP.

In the following sections parts of the general lake model represented by the DELTA model system LAKE will be described

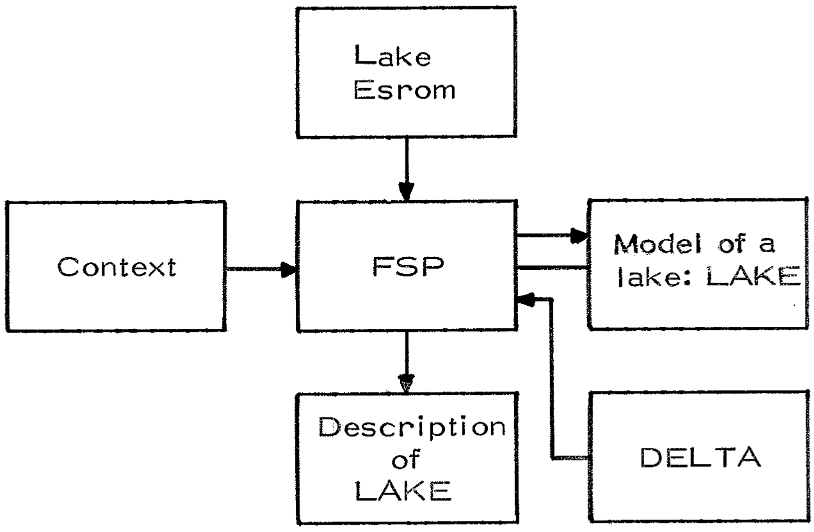


Fig. 1.5 The lake model LAKE

The figure above shows the "ideal situation", with the FSP group describing LAKE in DELTA. LAKE is described in co-operation with members of FSP (who had no knowledge of DELTA) and on the basis of FSP reports.

LAKE will be a composition of the models of the single descriplines. These are here represented by the DELTA model systems:

- TOPOGRAPHY/MORPHOGRAPHY LAKE
- HYDROLOGY/HYDRAULICS LAKE
- BIOLOGY LAKE
- MATHEMATICAL LAKE

A first description of the model system LAKE will be a description of which components, a lake is considered to contain and a first, sketchy description of the components.

The various groups will use this description as a frame of reference when describing the components further. We may say that the models of the disciplines are made in a context containing these basic lake concepts (fig. 1.6).

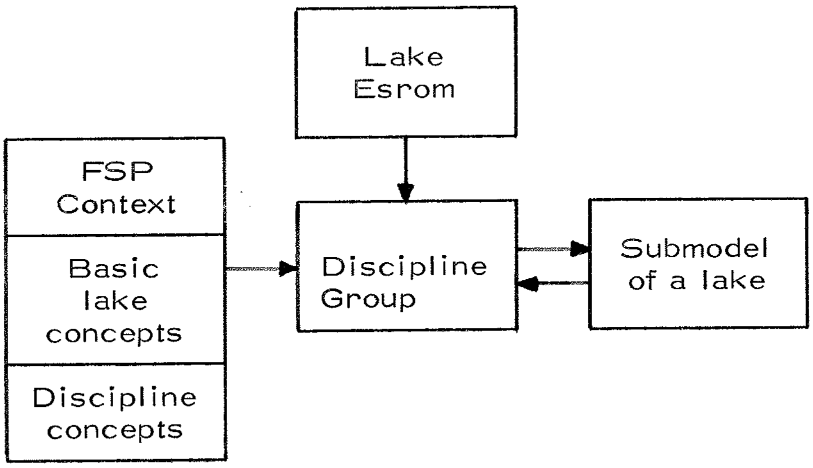


Fig. 1.6 The context of each group

1.2 Basic Lake Concepts

In this section there will be given a first description of the DELTA model system LAKE. This description is intended to act as a frame of reference for the groups describing different aspects of the lake. Therefore it is only a description of which components a lake is considered to consist of, and a first, not very detailled description of these. As referent system Lake Esrom may be considered, but LAKE is described on basis of knowledge of lakes in general. As mentioned before the model described here is not quite the same as the model in FSP; LAKE consists of more components than that of FSP.

A component of the referent system is in the DELTA model system represented by an object. In a DELTA system description an object is described by an object descriptor, which consists of a possible prefix (will be mentioned later) and a main part. The main part of an object descriptor is a text enclosed by the parantheses OBJECT BEGIN and END OBJECT.

The desctiption of a singular object is given by inserting the object name followed by a colon in front of the object descriptor, as e.g.:

A: OBJECT BEGIN ..... END SYSTEM

The referent system as a whole is represented by the DELTA system object, which is described by a DELTA system description:

S: SYSTEM BEGIN ..... END SYSTEM

The system object will contain the objects, which represent components of the referent system. If the referent system contains two components and if these are represented by the objects A and B , then the system object will contain A and B . This is described by:

```
S: SYSTEM BEGIN
       A: OBJECT BEGIN ..... END OBJECT;
       B: OBJECT BEGIN ..... END OBJECT
   END SYSTEM
```

A first description of the DELTA model system LAKE is

LAKE: SYSTEM BEGIN
an aquatic ecological system comprising the lake itself, the environment and those mechanisms, which are relevant in a ecological system
END SYSTEM

The next step is to identify relevant components of the system; this is called to decompose. A component may be decomposed in two ways:

- the component being decomposed is retained as a component of the model system and described as containing inner objects; this is called nesting

- the component being decomposed is replaced by a group of components. The decomposed component is then no longer a component of the model system. This way of decomposing is called splitting.

The first decomposition of a system object is always a nesting; the model system is said to consist of a number of components. These may by split by later decompositions.

A lake consists of the lake, the catchment area and the atmosphere over the lake and the catchment area. The catchment area is the area surrounding the lake and from where water is streaming to the lake and to where a permeation from the lake takes place.

The lake consists of the body of water (the water mass)and the sediment (fig. 1.7).

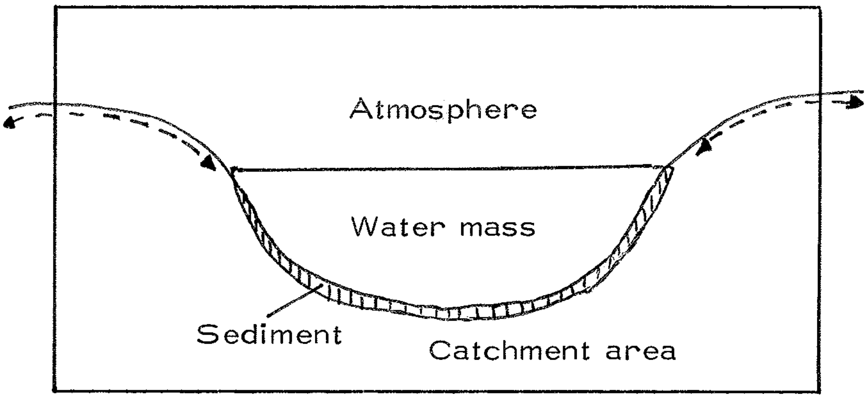
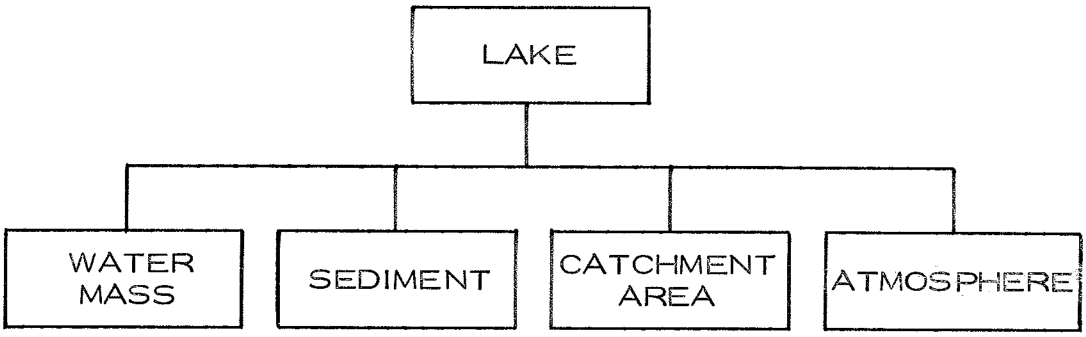


Fig. 1.7 The components of a lake

These components will in the model system be represented by the objects WATER MASS, SEDIMENT, CATCHMENT AREA and ATMOSPHERE.

The illustration of LAKE being decomposed into the objects by nesting is as follows; every object is illustrated by a rectangel containing the name of the object:



The system object is said to enclose the four objects, and these are said to be the content of the LAKE system object.

LAKE is a model of a lake. To distinquish between elements of the model and elements of the referent system the elements of the model will be denoted by capital letters, while elements of the referent system is denoted by small letters. That is, "atmosphere" denotes the physical existing atmosphere component of the referent system, while

"ATMOSPHERE" denotes the object of the modelsystem representing that component.

A first description of LAKE on this level of description is besides the first general description of the system object a description of the four objects:

```
LAKE: SYSTEM BEGIN
      WATER MASS: OBJECT BEGIN
                    the body of water with its content
                    of organic/inorganic materials and
                    biological/abiological processes
                  END OBJECT;
      SEDIMENT:   OBJECT BEGIN
                    the border layer between the water and
                    the seepage area, characterized by
                    instable temperatures and content of
                    inorganic compounds. Besides charac-
                    terized by processes influencing the
                    water
                  END OBJECT;
      CATCHMENT AREA:
                  OBJECT BEGIN
                    the soil environment, where water
                    courses, ground water and surface
                    flows influence the volume of the water
                    of the lake
                  END OBJECT;
      ATMOSPHERE:
                  OBJECT BEGIN
                    the atmospere over the water mass and
                    the catchment area, characterizing the
                    climate of the system (wind conditions,
                    rain fall and solar radiation)
                  END OBJECT
   END SYSTEM
```

The components identified here may in a later and more detailled description be decomposed further, until achieving components, which in some sense are elementary, or until a state is reached, where further decomposition would be irrelevant for the purpose.

In this first description of LAKE the WATER MASS and SEDIMENT will not be decomposed further. A later decomposition depends upon which aspects of the components are to be described. In a description of the exchange of materials between water and sediment it will be nessesary to decompose them, as it will be with the WATER MASS when to describe the hydraulic aspects of the lake.

The atmosphere is described by the <u>micro atmosphere</u> and the <u>macro atmosphere</u> (fig. 1.8).

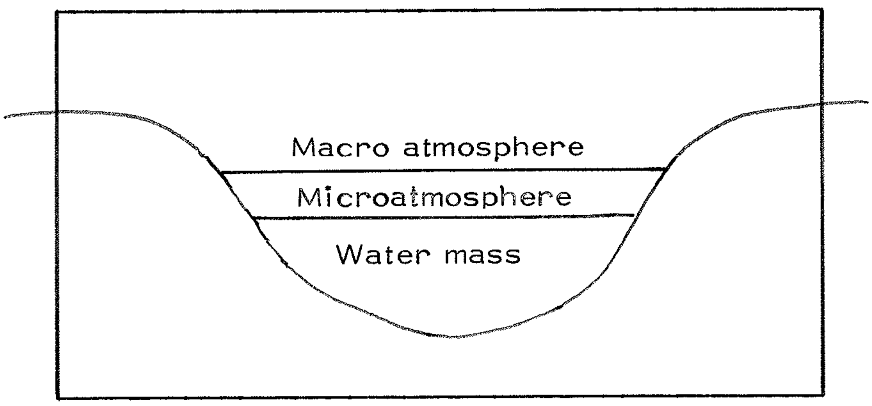
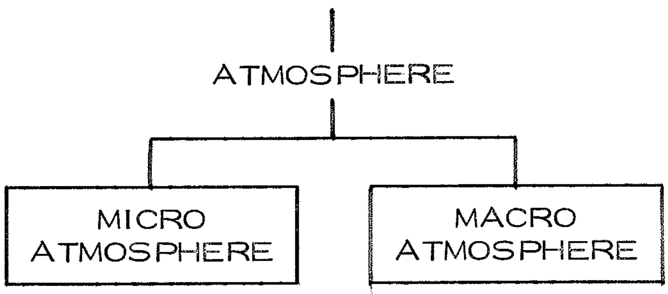


Fig. 1.8 Decomposition of the atmosphere

The atmosphere is described completely by these components, and the other components will interact with them independently and not as parts of an enclosing atmosphere component. This is reflected in the model by splitting ATMOSPHERE in the objects MICRO ATMOSPHERE and MACRO ATMOSPHERE:



By the decemposition the ATMOSPHERE object has been replaced by the two objects. This is illustrated by removing the rectangel surrounding "ATMOSPHERE".

A description of LAKE on this level of description consists of a description of the objects

WATER MASS, SEDIMENT and CATCHMENT AREA

as given before and a description of the objects MICRO ATMOSPHERE and MACRO ATMOSPHERE. This is indicated by:

```
LAKE: SYSTEM BEGIN
          :
        MICRO ATMOSPHERE:
          OBJECT BEGIN
            the part of the atmosphere right over the lake sur-
            face characterizing the wind conditions of the system
            and the exchange between air and water. Is influen-
            ced by the WATER MASS and the MACRO ATMOS-
            PHERE
          END OBJECT;
        MACRO ATMOSPHERE:
          OBJECT BEGIN
            the part of the atmosphere over the micro atmos-
            phere and the catchment area
          END OBJECT;
          :
      END SYSTEM
```

comment:

Here is missing a description of the aggregate ATMOSPHERE containing the objects MICRO ATMOSPHERE and MACRO ATMOSPHERE. In the first versions I used a notation like

ATMOSPHERE: AGGREGATE BEGIN
MICRO ATMOSPHERE: ....;
MACRO ATMOSPHERE: ....;
END AGGREGATE

but as an aggregate is no entity this is a bad solution. To illustrate that the objects have come into existence by splitting it is sufficient with a comment:

COMMENT ATMOSPHERE is an aggregate containing ......;

As I have had no need for a general aggregate concept, I have not suggested a way to describe an aggregate.
end comment

The catchment area is decomposed into surface, unsaturated layer and saturated layer (fig. 1.9).

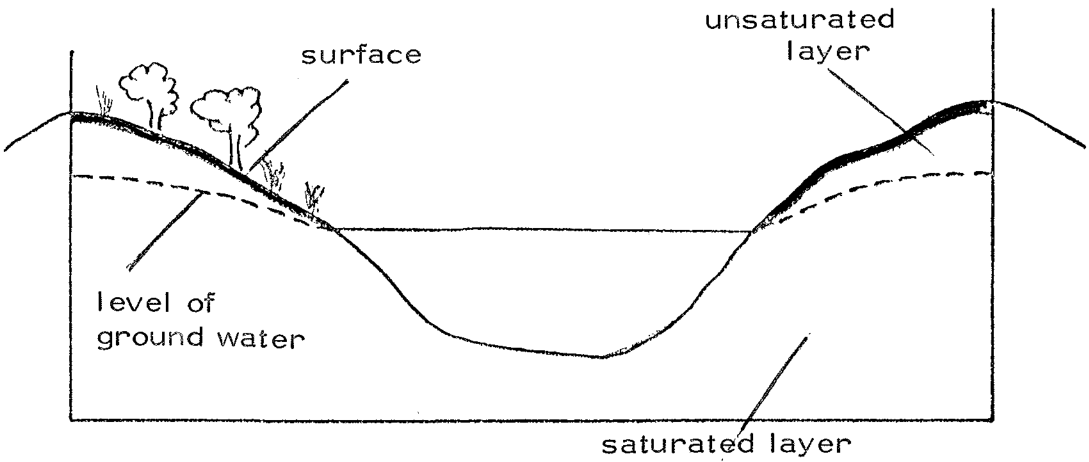
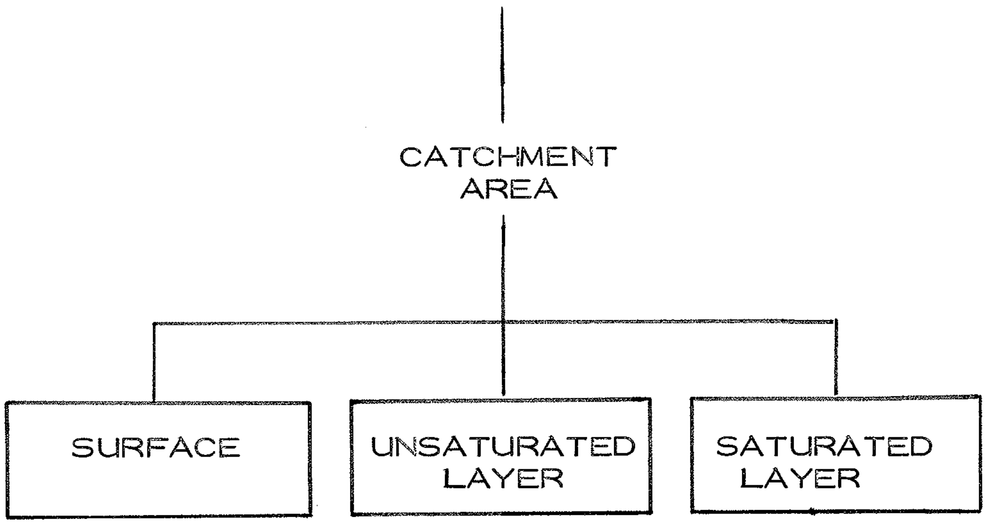


Fig. 1.9 The components of the catchment area

These components are in the model LAKE represented by the objects SURFACE, UNSATURATED LAYER and SATURATED LAYER. At this level of description the catchment area is completely described by these components and therefore does not have to be represented in the model system. Besides, the water and the sediment exchange directly with each of these components and not with them as parts of an enclosing catchment area component. This is in the model system reflected by splitting CATCHMENT AREA:

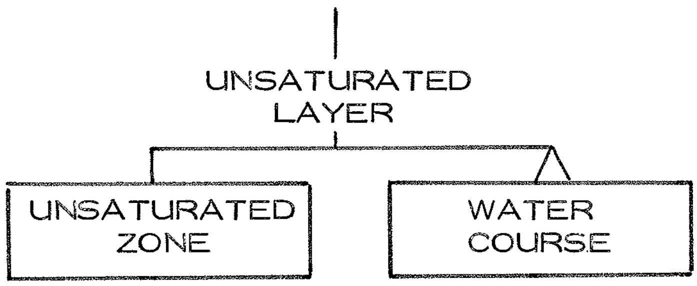


A further description of LAKE is a description of these objects:

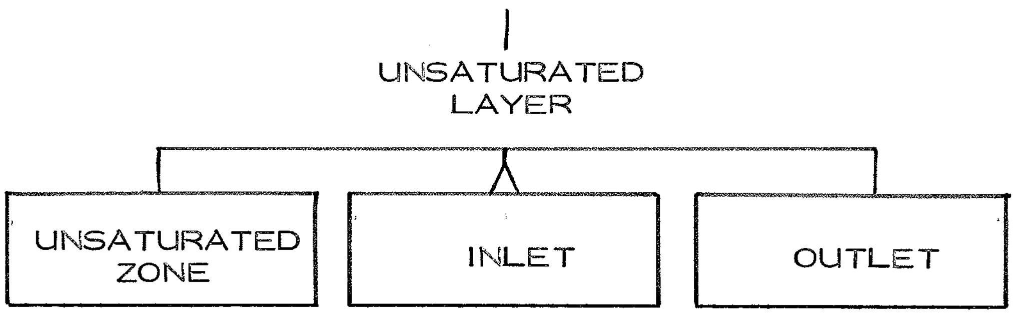
```
LAKE: SYSTEM BEGIN
            :
        SURFACE: OBJECT BEGIN
                    the upper layer of the catchment area with
                    higher plants and animals and from where
                    a surface flow of water to the lake may
                    take place
                 END OBJECT;
        UNSATURATED LAYER:
                 OBJECT BEGIN
                    the layer between the surface and the level
                    of ground water where the pores in the soil
                    are not entirely filled with water. The wa-
                    ter courses to and from the lake run in
                    this layer and a permeation inwards and
                    outwards takes place
                 END OBJECT;
        SATURATED LAYER:
                 OBJECT BEGIN
                    the layer from the level of the ground water
                    to a depth below the bottom of the lake;
```

```
        permeation of the ground water takes
        place from this layer
      END OBJECT;
   :
END SYSTEM
```

The saturated layer consists of the water courses and the area between the water courses, called the unsaturated zone. It is in the model represented by the object UNSATURATED ZONE. A water course is in the model represented by a WATER COURSE object. The unsaturated layer is completely described by the water courses and the unsaturated zone, and the components exchange directly with the water of the lake, so UNSATURATED LAYER is split. Normally there are more than one water course in connection with a lake, and the number of water courses is fixed as long as the lake is considered. This is illustrated by



There are two kinds of water courses: inlets, which lead water to the lake, and outlets, leading water from the lake. These components will in the model be represented by INLET and OUTLET objects. There will be more than one inlet and normally one outlet:



Even though an inlet and an outlet has different functions they are both water courses and therefore have common characteristics, as e.g. the "flow".

The common characteristics of water courses is in the model represented by WATER COURSE. While the WATER MASS and the SEDIMENT are singular objects (representing singular components of the referent system) WATER COURSE is a category of objects. In DELTA this is expressed by defining WATER COURSE as a class of objects:

CLASS WATER COURSE: OBJECT BEGIN...... END OBJECT

All objects of this class have the features described in the object descriptor.

INLET is a class of objects representing the inlets to the lake. An INLET object has the features of a WATER COURSE object plus some which specially characterize inlets. We say that INLET is a subclass of the class WATER COURSE, and this is in DELTA described by using the class title WATER COURSE as a prefix to the object descriptor:

```
CLASS INLET: WATER COURSE   OBJECT BEGIN
                                 :
                            END OBJECT
```

OUTLET is a singular object with elements of WATER COURSE objects plus elements specially characterizing an outlet:

OUTLET: WATER COURSE OBJECT BEGIN...... END OBJECT

Provided with these concepts a description of UNSATURATED LAYER as split in UNSATURATED ZONE, INLETs and OUTLET is given by:

```
LAKE: SYSTEM BEGIN
        :
        UNSATURATED ZONE: OBJECT BEGIN
                                    the unsaturated layer be-
                                    tween the water courses of
                                    the lake and from/to where
                                    a permeation inwards/out-
                                    wards takes place
                                   END OBJECT;
        CLASS WATER COURSE: OBJECT BEGIN...END OBJECT;
        CLASS INLET:
          WATER COURSE  OBJECT BEGIN
                                    water courses leading
                                    water into the lake
                                 END OBJECT;
        OUTLET : WATER COURSE OBJECT BEGIN
                                      the water course leading
                                      water from the lake
                                     END OBJECT;
        :
      END SYSTEM
```

The components identified now is considered being the basic components of a lake. As a frame of reference for the various groups the LAKE consists of the following objects:

1.3 Topographic/morphographic description of a lake

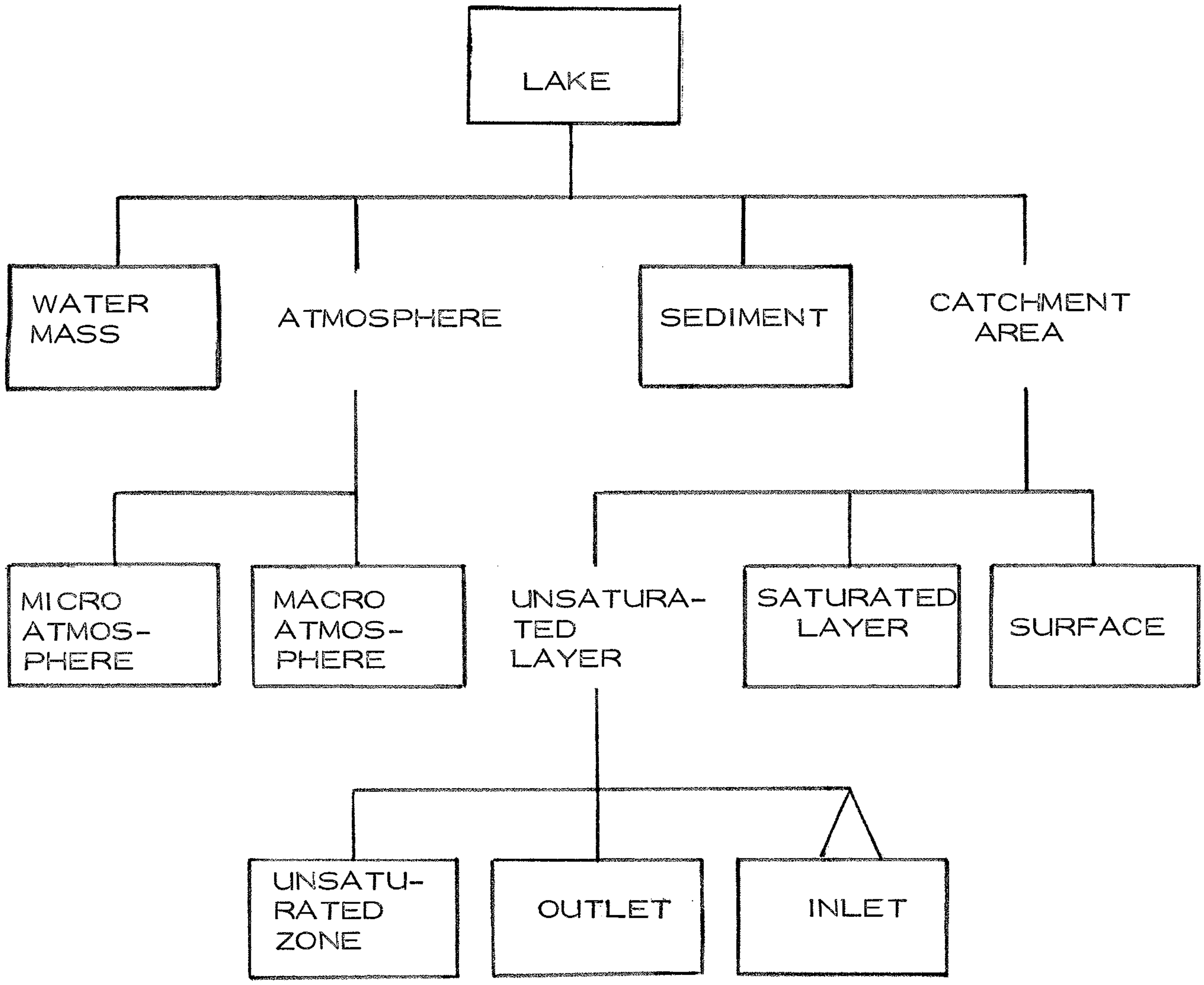
The topography and morphography of a lake locates the lake in a wider ecosystem and tells something about the size and shape of the lake and about the nature of the catchment area. With the description of LAKE in section 1.2 as a frame of reference the following is a description of the topographic/morphographic aspects of the components of a lake.

Every component of the referent system is associated with a number of quantities, characterizing the component. The lake as a whole is characterized by position and altitude, the catchment area by population density and sewage load, and the water by whether it is used for fishing, recreation and collection of drinking water or not.

The quantities associated with a component are said to be part of the attributes of the component. Other attributes are references and patterns.

A quantity is the association of a name, a type indicating a way of measuring something (e.g. amount or quality) and a value. The type defines a set of possible states, which may be observed by the measuring, and the possible operations upon these states. The value of a quantity is one of these states. The values of the quantities of a component at a given moment form part of the state of the component.

References are introduced later. A pattern is a definition of a category of system elements with common features. An example of a category defined by a pattern is a class, defining a category of objects. It defines the permanent properties of all objects of the class, but each object is an independent element with its own state.


LAKE
WATER MASS
ATMOSPHERE
SEDIMENT
CATCHMENT AREA
MICRO ATMOS-PHERE
MACRO ATMOS-PHERE
UNSATURA-TED LAYER
SATURATED LAYER
SURFACE
UNSATU-RATED ZONE
OUTLET
INLET

The attributes of a component are represented by the attributes of the corresponding object. Quantities are represented by symbols, giving the name, type and value (at a given moment), while patterns are represented by descriptions of these. The LAKE object will as an attribute have e.g. a description of the class of INLET objects:

CLASS INLET: WATER COURSE OBJECT BEGIN. . . END OBJECT.

The various INLET objects of the model system will be introduced according to this class declaration.

An object descriptor contains a description of the attributes of the object:

```
OBJECT BEGIN
  description of the
  attributes of the object;
  :
END OBJECT.
```

The value of a quantity may vary through its existence or it may be constant. In the former case it is said to be a variable, in the latter case a constant. Some types are given in the DELTA language, as e.g. INTEGER, REAL and BOOLEAN (will be mentioned later). Descriptions of quantities with these types have the form:

for variables: X, Y, ...: INTEGER
for constants: A, B, ...: INTEGER CONSTANT

The position of the lake is measured in

East longitude, North latitude,

and these quantities are both measured in degrees and minutes. A type defining measurement in (East longitude, North latitude) is not given by the DELTA language. The quantity "position" is therefore described by giving the specification of a measure with the desired properties. This specification consists of a text enclosed by the brackets MEASURE BEGIN and END MEASURE. A description of the quantity is thus:

```
POSITION: MEASURE BEGIN
              position measured in
              East longitude, North latitude
          END MEASURE
```

A more detailed description of POSITION will be to describe it by two quantities: EAST LONGITUDE, NORTH LATITUDE. These quantities would be of the same type, geographical measure, which is measured by two integers: degrees and minutes. This type is not given in the DELTA language, but may be described by a type declaration:

```
TYPE GEOGR MEASURE:
        MEASURE BEGIN
           DEGREES, MINUTES: INTEGER;
           description of geographical measure,
           e. g. of operations on values of this type
        END MEASURE
```

A type is a pattern defining a category of measures and gives thus the common features of quantities of this type. Every quantity with the type GEOGR MEASURE has thus a pair of integers as value.

INTEGER is a type defined in DELTA. Quantities of this type may have the values

0, + 1, - 1, + 2, - 2, ...

and to the type is associated the usual operations +, -, / and *, which may be applied to the values of the type. By

```
DEGREES, MINUTES: INTEGER
```

two quantities with the type INTEGER and the names "DEGREES" and "MINUTES" are introduced.

A more detailed description of GEOGR MEASURE will show that it has associated other operations than the usual. Addition will thus be defined in another way than integer addition. If the values of DEGREES and MINUTES are 55 and 60 respectively, then addition of 1 to MINUTES will not give the result (55, 61) but (56, 0).

For types, which are not part of the DELTA language, description of quantities with these types has the form:

for variables: X, Y, ...: <u>QUANTITY</u> <u>OF</u> GEOGR MEASURE
for constants: A, B, ...: <u>CONSTANT</u> <u>QUANTITY</u> <u>OF</u> GEOGR MEASURE

POSITION may thus be described by:

POSITION: <u>MEASURE</u> <u>BEGIN</u>
EAST LONGITUDE,
NORTH LATITUDE: <u>CONSTANT</u> <u>QUANTITY</u> <u>OF</u> GEOGR MEASURE
<u>END</u> <u>MEASURE</u>

as the position of a lake is constant. The value of POSITION is the pair of values of EAST LONGITUDE and NORTH LATITUDE.

<u>comment:</u>

Descriptions of measures and types are not included in DELTA as it is given now, so the preceding shall be seen as a proposal, but only on the descriptive level.

To avoid the difference between declaration of quantities with language-defined types and quantities with other types there are two possibilities:

1. Do not regard INTEGER, REAL, ... as special types, but as types, which may be defined by the generalized type concept. The declaration of a INTEGER variable should then be

I: QUANTITY OF INTEGER

2. Use the notation of language-defined types to declare quantities of other types, e.g.

X: GEOGR MEASURE

The last possibility implies that the set of keywords may be extented in the specification of a system. The first possibility is conceptually in accordance with the idea of a generalized type concept and should thus be preferred. But, in cases it will be a cumbersome notation.
end comment

The attitude of the lake is measured in height ± water level variation. So it is described by two quantities of the type linear measure, which besides a real number is described by an unit of measurement and special operations including rules for conversions from one unit of measurement to another. Linear measure is said to be a subtype to the type REAL. In DELTA this is described by

```
TYPE LINEAR MEASURE:
        REAL MEASURE BEGIN
                description of linear measuring
                with associated unit of
                measurement and operations
             END MEASURE
```

The type REAL is part of the DELTA language. Quantities of this type may have as values the real numbers, and to the type is associated the usual operations +, -, /, *. In a subtype there may be defined alternative operations or extensions of those given in the type. The value of a quantity of type LINEAR MEASURE is: a real number and the value of the unit of measurement.

Provided with the type LINEAR MEASURE the ALTITUDE may be described:

```
ALTITUDE: MEASURE BEGIN
              HEIGHT: CONSTANT QUANTITY OF LINEAR
                                          MEASURE;
              WATER LEV VAR: QUANTITY OF LINEAR
                                          MEASURE;
              is measured in
                 HEIGHT ± WATER LEV VAR;
              unit of measurement is "meter"
          END MEASURE
```

The use of the lake for e.g. fishing may be characterized by it being used for fishing or not being used for fishing. To describe this the type BOOLEAN may be used. Quantities of this type may have as value TRUE and FALSE, and to the type are associated the usual logical operations. The use of the lake for fishing may thus be described by a quantity FISHING:

```
FISHING: BOOLEAN
```

Further the lake is characterized by two boolean quantities telling whether it is used for recreation and collection of drinking water.

A component is besides quantities, references and patterns characterized by being able to execute actions. The actions of a component consist of an action sequence. Parts of this sequence may be executed according to action patterns, which may be one of the attributes of the object or of other objects. They define possible sequences of actions, which the object may execute or impose upon other objects.

Apart from the attributes an object is characterized by its own task: a sequence of actions, the object will start to execute when introduced into the DELTA system.

An object decriptor is thus:

```
OBJECT BEGIN
    description of
    attributes
    OWN TASK BEGIN
        description of a
        sequence of actions
    END TASK
END OBJECT        (note below)
```

The system object descriptor has the same form.

From the point of view of topography and morphography the system and its components are not characterized by executing any actions. So the own tasks of the objects are omitted:

```
TOPOGRAPHY/MORPHOGRAPHY LAKE:
   SYSTEM BEGIN
      TYPE GEOGR MEASURE: .....;
      TYPE LINEAR MEASURE: ....;
      POSITION: .....;
      ALTITUDE: ....;
      WATER MASS:
         OBJECT BEGIN
            FISHING, RECREATION: BOOLEAN
         END OBJECT;
      :
   END SYSTEM
```

Notice that the single system description elements are separated by semicolons (;).

(note)

In the present version of the DELTA language the task associated with an object is called its prime task. The keyword OWN should then read PRIME.

The population density and the sewage load characterize the surface component of the catchment area. The population density is measured by an integer indicating number of persons per km^2. The sewage load is measured in B.O.D. (biochemical oxygen demand) per every sewage treatment plant in the catchment area.

```
TYPE B.O.D.: MEASURE BEGIN
                measure of sewage load
             END MEASURE
```

The load is represented by a number of quantities of the type B.O.D., one quantity for every plant. This is described by:

```
LOAD: ARRAY (1: NUMBER OF PLANTS) OF B.O.D.,
```

where NUMBER OF PLANTS is an integer quantity.This describes the quantities

```
LOAD (1), LOAD (2), ....., LOAD (NUMBER OF PLANTS).
```

The total load would be

```
LOAD (1) + LOAD (2) + ..... + LOAD (NUMBER OF PLANTS)
```

comment:

An array concept is not part of the DELTA, so I have here used a combination of SIMULA-notation and notation for declaration of quantities. This because I have had no need of a detailed array concept.

end comment

Besides the already mentioned characteristics, the surface of the catchment area is characterized by it being used for recreation or for agriculture. A further topographic description of LAKE is thus:

```
TOPOGRAPHY/MORPHOGRAPHY LAKE:
   SYSTEM BEGIN
     :
     SURFACE:
       OBJECT BEGIN
         NUMBER OF PLANTS: INTEGER;
         TYPE B. O. D: .......... ;
         LOAD: ......... ;
         POP DENSITY: INTEGER MEASURE BEGIN ........
                                      END MEASURE ;
         RECREATION, AGRICULT: BOOLEAN
       END OBJECT;
     :
   END SYSTEM
```

The morphography of a lake is described by the area of the catchment area, the area of the lake surface, the maximum depth, the average depth and the volume of the water. The shape of the lake basin is described by the hypsographic curve, which for every depth "d" yields the area of a horizontal flake positioned in the distance "d" from the lake surface (fig. 1.10).

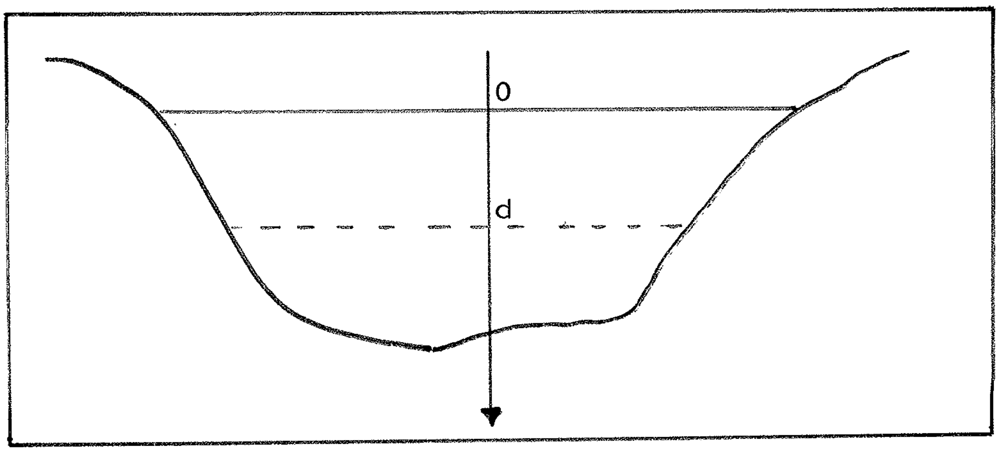


Fig. 1.10

On the basis of this a depth-volume curve may be computed. This gives for a depth "d" the volume of the body of water below "d" (fig. 1.11).

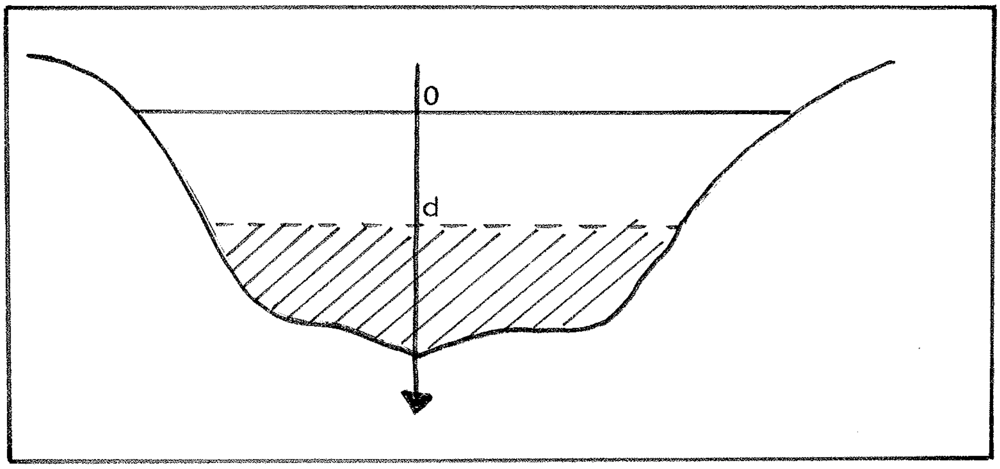


Fig. 1.11

These curves depend upon the sedimentation and therefore upon time, but the sedimentation is in this context insignificant, so the curves are considered only to depend upon the depth. The curves are in the model represented by the functions HYPSOGRAPH and DEPTH-VOLUME, so that

area = HYPSOGRAPH (depth)
volume = DEPTH-VOLUME (depth).

Functions are in the DELTA language described by a function declaration:

FUNCTION F: <type> BEGIN .... END

"F" is the title of the function. The value of the function at a given moment is the result of the evaluation described in BEGIN ...... END, and it may be one of the values given by the <type> . A function taking integer values is thus described by

FUNCTION G: INTEGER BEGIN .... END

Parameters may be specified by:

FUNCTION HYPSOGRAPH (DEPTH): BEGIN ....END.

The result of an evaluation is the value of a predefined quantity RESULT. Actions of the evaluation may assign a value to RESULT.

A function is an example of an action pattern - defining a category of evaluations. The action sequence of the evaluations of such a category is described in the evaluation descriptor "BEGIN ..... END" of the function declaration. When a function title is encountered in evaluation of an expression, then an action sequence following the evaluation descriptor is executed. The same function may be used in different contexts, yielding different results, dependent upon e.g. the values of parameters.

The two curves are different for different lakes. In a general lake model it can only be stated that it is characterized morphographically by the two functions. To express that they are undefined at this level of description they are introduced as virtual elements:

VIRTUAL: FUNCTION HYPSOGRAPH, DEPTH-VOLUME.
(note below)

In this case the evaluation descriptors are omitted.

The morphography of a lake is further determined by maximum length, maximum with and maximum length of the coastline. The type LINEAR MEASURE is introduced for description of quantities. For description of quantities for measuring areas and volumes are the types AREA MEASURE and VOLUME MEASURE introduced:

TYPE AREA MEASURE: REAL MEASURE BEGIN
measuring of areas with
unit of measuring = m^2
END MEASURE

(note)

In the present DELTA this would be written:

FUNCTION HYPSOGRAPH, DEPTH-VOLUME : VIRTUAL.

```
TYPE VOLUME MEASURE:   REAL   MEASURE BEGIN
                                measuring of volumes with
                                unit of measuring = m3
                              END MEASURE
```

So the lake model is further described by:

```
TOPOGRAPHY/MORPHOGRAPHY LAKE:

    SYSTEM BEGIN
      :
      TYPE LINEAR MEASURE: .....;
      TYPE AREA MEASURE: .......;
      TYPE VOLUME MEASURE: ....;
      MAX LENGTH, MAX WIDTH, SHORE LENGTH:
                                  QUANTITY OF LINEAR MEASURE;
      SURFACE: OBJECT BEGIN
                 :
                 AREA: QUANTITY OF AREA MEASURE;
                 :
               END OBJECT;
      VIRTUAL: FUNCTION HYPSOGRAPH, DEPTH-VOLUME;
      WATER MASS:
          OBJECT BEGIN
            :
            MAX DEPTH, AVERAGE DEPTH: QUANTITY OF
                                         LINEAR MEASURE;
            AREA: QUANTITY OF AREA MEASURE;
            VOLUME: QUANTITY OF VOLUME MEASURE;
            :
          END OBJECT;
      :
    END SYSTEM
```

When nothing else is stated quantities of type INTEGER will get the initial value 0, quantities of type REAL 0.0 and BOOLEANs will get FALSE. As described now the quantity POSITION of the DELTA model system TOPOGRAPHY/MORPHOGRAPHY LAKE will get the initial value ((0,0), (0,0)). The starting point of a DELTA model system is determined by the system generator. From the moment the system generator starts to generate a model system by interpreting a system description the system object and its content of objects exist.

A topographic/morphographic description of a specific lake, e.g. Lake Esrom, will be the description as given of TOPOGRAPHY/ MORPHOGRAPHY LAKE, but with the values of the quantities which are valid for Lake Esrom.

This is achieved by assigning the appropriate values to the quantities when the system is generated. The description of this is:

```
TOPOGRAPHY/MORPHOGRAPHY LAKE:

    SYSTEM BEGIN
    .
    .
    .
    END SYSTEM

      PUT (* POSITION: = ((12,22) , (56,0));
             ALTITUDE: = (0, 0.3);
             .
             .
             .
          *)
```

All quantities first get the initial values according to their type. Then the assignments described in PUT (* .... *) are performed.

comment

Note the cautious use of initializations in PUT (* .... *).
The problem is that it is not possible to initialize internal, singular objects as e.g. the WATER MASS and the SURFACE. This is disadvantageous, especially in the case where a system class is described.

end comment

As mentioned the goal is to make a general lake model. LAKE is thus considered as a pattern for models of specific lakes. In the same way TOPOGRAPHY/MORPHOGRAPHY LAKE is thus considered as a pattern for topographic/morphographic models of lakes and not as a singular model system.

From that point of view TOPOGRAPHY/MORPHOGRAPHY LAKE is a class of DELTA system objects, in the same way as WATER COURSE in section 1.2 is a class of objects. In the existing version of DELTA the concept "class of DELTA system objects" is not given, but only "class of objects". The introduction of a concept "class of system objects" will be discussed in the coming report on contexts.

On the assumption that a "system class" concept is given a topographic/ morphographic model of lake Esrom will be a DELTA model system with the elements described in the class of systems TOPOGRAPHY/ MORPHOGRAPHY LAKE. In the description of the model of Lake Esrom, T/M LAKE ESROM, the title "TOPOGRAPHY/MORPHOGRAPHY LAKE" will be used as prefix (as was the title "WATER COURSE" in the description of the singular object OUTLET in section 1.2), thus indicating that T/M LAKE ESROM is a DELTA system object of the class TOPOGRAPHY/ MORPHOGRAPHY LAKE:

```
T/M LAKE ESROM:
     TOPOGRAPHY/MORPHOGRAPHY LAKE SYSTEM BEGIN
                                    .
                                    .
                                    .
                                  END SYSTEM
                                     PUT (*.....*)
```

The DELTA model system T/M LAKE ESROM will thus consist of the elements of TOPOGRAPHY/MORPHOGRAPHY LAKE plus the elements described in SYSTEM BEGIN .....END SYSTEM, and with the initial values as stated in PUT (*.....*).

For Lake Esrom the functions HYPSOGRAPH and DEPTH-VOLUME are known. They may therefore be defined in T/M LAKE ESROM and thus replace the virtual elements of TOPOGRAPHY/MORPHOGRAPHY LAKE:

```
T/M LAKE ESROM:
  TOPOGRAPHY/MORPHOGRAPHY LAKE
    SYSTEM BEGIN
      FUNCTION HYPSOGRAPH (DEPTH):
        AREA MEASURE
          BEGIN
            DEPTH: QUANTITY OF LINEAR MEASURE;
            RESULT: =
              IF DEPTH ≤ 15 THEN (17.253 - 1.142 * DEPTH
                                 + 0.066*DEPTH^2 - 0.0018^3)
              ELSE
              IF DEPTH > 15 AND DEPTH ≤ 20 THEN
                                      (8.9 - (DEPTH-15)* 0.98)
              ELSE
              IF DEPTH ≤ MAX DEPTH THEN (4.0 - (DEPTH-20)* 2)
          END HYPSOGRAPH;

      FUNCTION DEPTH-VOLUME:
        VOLUME MEASURE
          BEGIN
            DEPTH : QUANTITY OF LINEAR MEASURE ;
            RESULT : = function of DEPTH and HYPSOGRAPH(DEPTH)
          END

    END SYSTEM
```

1.4 Hydrological/hydraulic Aspects of a Lake

In the following the DELTA model system HYDROLOGY/HYDRAULICS LAKE will be described. It will be a description of the hydrological and hydraulic aspects of the components identified in section 1.2. The component in focus is the WATER MASS of the lake. The exchange of water between the WATER MASS and the other components and the mixing of the water of the water mass are described.

The mixing of water is brought about by currents. The most important of these are the turbulent (or disordered) currents, which may be due to differences in density ("density" is mentioned later) and influence of the wind. Among these the wind is the most important.

A pure hydraulic description of a lake would be a description of the water mass as one component characterized by a number of different currents, the velocity and direction of which would vary as time. The description given here is supposed to be used together with a biological description, and as this is made most easily by dividing the water mass in a fixed number of parts (layers), the following hydraulic description will be made according to this. But it shall be noted that with respect to hydraulics it is a rigid system and not a pure hydraulic description. With the water mass divided in layers the mixing is described by exchange of water between the layers.

Division of the water mass in layers

When dividing the water mass in a number of fixed layers of water, e.g. as in Figure 1.12, it is assumed that the single layer is entirely mixed, that is there are no (or only small) horizontal gradients (differences in content or concentration) and the layers are chosen so small that the gradient from the surface to the undersurface of the layer is small. This means that the dividing into layers of a specific lake must be done at a time where large gradients in the whole water mass are given, e.g. when the lake is stratified (this phenomenon is mentioned later).

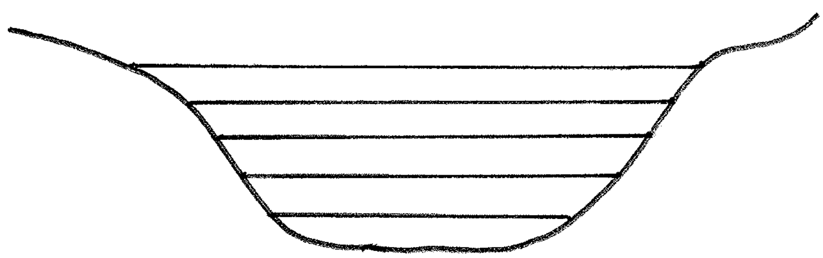


Fig. 1.12

Common to all water layers is that they are supplied with water and that water is removed from the layer. This is due to external processes (e.g. inflow, rain fall, outflow) and internal, that is by mixing.

The supply and removal are measured in volume of water per unit of time. Measurement of this kind will be described by the type RATE OF FLOW. Quantities of this type consists of a real number, indicating the volume of water, and a number indicating the unit of time.

```
TYPE RATE OF FLOW: REAL MEASURE BEGIN
                        measurement of flow of water as
                        volume per unit of time
                        END MEASURE
```

To the type are associated the usual operations of REAL extended with possibilities of conversion from one unit of time to another.

The class WATER LAYER

The common features of all the water layers are in the model represented by the class WATER LAYER.

For the description of WATER LAYER it would be appropriate to be able to use the measure concepts LINEAR MEASURE, AREA MEASURE and VOLUME MEASURE as introduced in section 1.3. A way of indicating this would be to collect them, under the name TOPOGRAPHY/MORPHOGRAPHY, and use this name as a prefix in the system description:

```
HYDROLOGY/HYDRAULICS LAKE:
TOPOGRAPHY/MORPHOGRAPHY
  SYSTEM BEGIN
     TYPE RATE OF FLOW: .....;
     LAYERS ): number of layers: INTEGER;
     CLASS WATER LAYER:
          OBJECT BEGIN
               DEPTH: QUANTITY OF LINEAR MEASURE;
               SED SURFACE ): surface of contact with the
               sediment: QUANTITY OF AREA MEASURE;
```

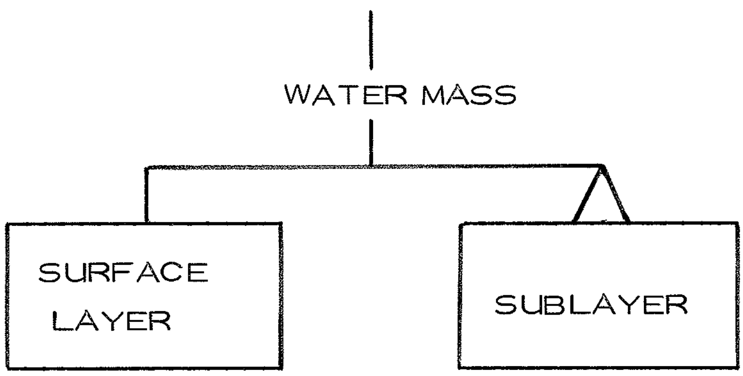
```
            VIRTUAL : FUNCTION WATER SUPPLY, WATER REMOVAL;
            these are both of the type RATE OF FLOW;
            :
        END OBJECT ;
    :
    END SYSTEM
```

WATER SUPPLY and WATER REMOVAL are virtuals, as their definition depend upon whether the layer is the surface layer, the bottom layer or a layer between these.

Note that in describing a quantity an explanatory comment may be inserted after the name of the quantity, as in "LAYERS): number of layers : INTEGER".

Decomposition of WATER MASS

All layers are supplied with water from inlets, permeation inwards and from the surrounding layers by mixing, and water is removed from all layers by permeation outwards and by mixing. Rain fall, surface streaming, the outlet and the evaporation influence only the surface layer of the lake. So in a first decomposition of the water mass are identified the components surface layer and the layers below this (sublayers). These are in the model represented by the object SURFACE LAYER and the class SUBLAYER. The WATER MASS is split because the other objects interact directly with the SURFACE LAYER and the SUBLAYER objects:



The definitions of WATER SUPPLY and WATER REMOVAL are different in the two kinds of layers:

```
HYDROLOGY/HYDRAULICS LAKE:
    TOPOGRAPHY/MORPHOGRAPHY
        SYSTEM BEGIN
        :
        CLASS WATER LAYER: OBJECT BEGIN
                                    :
                                    INFLOW ): water from inlets, PERMEA-
                                    TION IN, PERMEATION OUT, MIX
                                    SUPPLY ): supply of water by mixing,
                                    MIX REMOVAL:
                                        QUANTITY OF RATE OF FLOW;
                                    :
                                    END OBJECT;
        SURFACE LAYER:
            WATER LAYER OBJECT BEGIN
                                PRECIPITATION, STREAMING ): from the
                                SURFACE of the CATCHMENT AREA,
                                OUTFLOW ): water removal by the outlet,
                                EVAPORATION : QUANTITY OF RATE OF FLOW;
                                FUNCTION WATER SUPPLY:
                                    RATE OF FLOW
                                        BEGIN
                                            RESULT:= INFLOW + PERMEATION
                                            IN + PRECIPITATION +
                                            STREAMING + MIX SUPPLY
                                        END;
                                FUNCTION WATER REMOVAL:
                                    RATE OF FLOW
                                        BEGIN
                                            RESULT:= OUTFLOW + PERMEA-
                                            TION OUT + EVAPORATION +
                                            MIX REMOVAL
                                        END;
                                :
                                END SURFACE LAYER OBJECT
```

```
CLASS SUBLAYER :
  WATER LAYER OBJECT BEGIN
                     FUNCTION WATER SUPPLY :
                       RATE OF FLOW BEGIN
                                    RESULT :=
                                      INFLOW + PERMEATION IN
                                      + MIX SUPPLY
                                  END ;
                     FUNCTION WATER REMOVAL :
                       RATE OF FLOW BEGIN
                                    RESULT := PERMEATION OUT
                                              + MIX REMOVAL
                                  END ;
                     :
                  END SUBLAYER OBJECT
:
END SYSTEM
```

A hydraulic description will now be a description of the contributions MIX SUPPLY and MIX REMOVAL. The hydrological aspects involve the remaining contributions.

1.4.1 The hydrological Aspects

The volume of the single layer is assumed to be constant, so that WATER SUPPLY = WATER REMOVAL.

Permeation

The nature of the permeation of water is not known as well as the other contributions, so at this level of description it is undefined:

```
CLASS WATER LAYER:
  OBJECT BEGIN
    :
    FUNCTION PERMEATION IN, PERMEATION OUT ;
    dependent upon DEPTH, SED SURFACE and the conditions
    of SATURATED LAYER and UNSATURATED ZONE ;
    :
  END OBJECT
```

Inflow

All water layers may get water from the inlets as the water is imbedded into the layer with the same temperature and density as the water from the inlets.

The amount of water from the inlets is determined by the flow for each of the inlets. As the concept "flow" is common to inlets and outlets it is characterizing the class of WATER COURSE objects. The flow is measured in litre per second and is thus of type RATE OF FLOW. The flow is determined on basis of measurings taken e.g. every 14 days, by interpolating and eventually converting to the unit of time used in the system as a whole (e.g. litre per day).

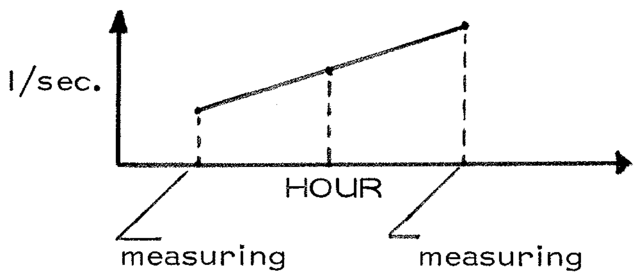


Fig. 1.13 FLOW

```
CLASS WATER COURSE :
  OBJECT BEGIN
      NUMBER OF MEASURINGS : INTEGER ;
      MEASURINGS : ARRAY (1: NUMBER OF MEASURINGS)
        OF RATE OF FLOW ;
      FUNCTION FLOW :
          RATE OF FLOW BEGIN
                          determine RESULT by interpolation
                          between the last and the next measuring
                        END ;
      END OBJECT
```

The value of INFLOW is for each WATER LAYER determined as the sum of the flows of the inlets contributing to the layer. These inlets are denoted by a number of references:

INLETS : ARRAY (1: NUMBER OF INLETs) OF REF INLET.

Singular objects, as e.g. SURFACE LAYER, are denoted by their names. Class-defined objects cannot be denoted by the class title, so the DELTA language contains references, the function of which is to denote or to refer to objects. A reference is an association of a name, a qualification, giving the class of objects which may be denoted, and a value, which is an object of this class. A reference is described by

<name> : REF <class title> .

In determining the value of INFLOW the values of the FLOW quantities of the single INLET objects must be available. This is obtained by the construct

INLETS (I). FLOW

where INLETS (I) denotes one of the INLET objects. The dot (".") is a substitution of the genetive "'s".

The value of INFLOW is determined by examining the INLETs one after another and summing up the FLOWs of the INLETs having the same DENSITY as the layer:

```
:
NUMBER OF INLETS : INTEGER ;
INLETS : ARRAY (1: NUMBER OF INLETS) OF REF INLET;
CLASS WATERLLAYER :
  OBJECT BEGIN
    :
    TEMP ): temperature, DENSITY : REAL ;
    FUNCTION INFLOW :
      RATE OF FLOW BEGIN
                        I : INTEGER ;
                        RESULT := 0 ;
```

```
                    FOR I := 1 STEP 1 UNTIL NUMBER OF INLETS
                      REPEAT
                        (* IF INLETS (I). DENSITY = DENSITY
                           THEN RESULT := RESULT + INLETS(I).
                        *)                                  FLOW
                    END ;
          .
          .
          .
          END OBJECT ;
        .
        .
        .
```

The imperative "FOR I := ... REPEAT (* ..... *)" describes that the action sequence described in "(* ..... *)" shall be executed repeatedly, with I taking the values 1, 2, 3, ..., NUMBER OF INLETS.

Precipitation

The precipitation of the model contributes only to the water mass and not to the catchment area. The rain fall over this area contributes indirectly through surface streaming, inflowing and permeation inwards.

The precipitation depends upon the time (of the year). It is determined on the basis of measurings at different positions in the system, and it is measured in mm per area per day. For the hydrological description it is sufficient to measure the recipitation in mm per area per month, so the PRECIPITATION is described by a periodic function, with the period 12 and constant for every period (month) (Fig. 1.14).

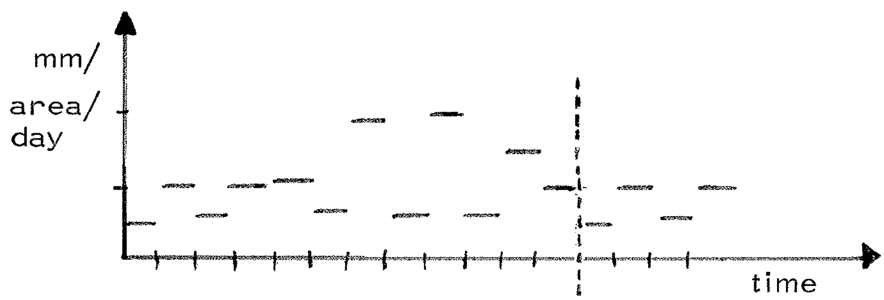


Fig. 1.14. PRECIPITATION

The PRECIPITATION in a given month is determined as the mean value of measurings at different positions in the system in the actual month:

```
ATMOSPHERE :
 OBJECT BEGIN
      NUMBER OF MEASURINGS : INTEGER ;
      TYPE RAIN GAUGING : REAL MEASURE BEGIN
                                         measuring of precipitation in
                                         mm/area/time
                                       END MEASURE ;
     MEASURINGS : ARRAY (1: NUMBER OF MEASURINGS ,
                               1: 12) OF RAIN GAUGING ;
     FUNCTION PRECIPITATION:
          RAIN GAUGING BEGIN
                               determine RESULT as the mean value
                               of MEASURINGS(i,month),
                               i = 1,2,3, ... NUMBER OF MEASURINGS
                             END ;
  .
  .
  .
 END ATMOSPHERE OBJECT
```

In DELTA model systems the time is represented by the real quantity TIME, the value of which increases continuously. The value of PRECIPITATION thus depends upon the value of TIME. When PRECIPITATION is encountered in an evaluation of e.g. the WATER SUPPLY of the SURFACE LAYER:

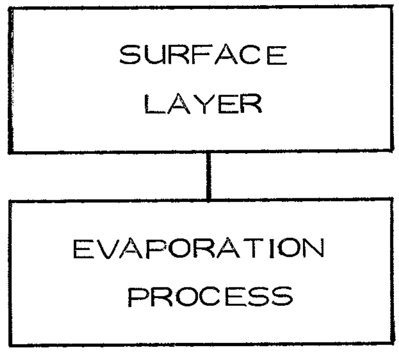
RESULT := INFLOW + ... + RECIPITATION + ...

then it yields as result the mean value of the MEASURINGS of the month, in which the evaluation of WATER SUPPLY is made, that is of the month indicated by the value of TIME.

Evaporation

The evaporation from the surface of the water mass is determined either by measurings within and outside the lake system or by making a heat balance for the lake, with the contribution of the evaporation as unknown. The evaporation is in the same way as the recipitation measured in mm per area per month.

The evaporation is a process contained in the surface layer of the water mass; the process defines the amount of water disappearing from the layer. As mentioned earlier an object may execute a sequence of actions. The process of evaporation may then be represented by an internal object EVAPORATION PROCESS:



the action of which defines the value of the quantity EVAPORATION of the SURFACE LAYER.

In DELTA an action is an instantaneous action or a time consuming action. Instantaneous actions are actions which are executed without having the value of the model time TIME increased. Note that TIME is the model time. In a physically existing referent system there are no instantaneous actions, but there are actions which we in a model of the system want to regard as instantaneous.

Actions in the DELTA language are described by imperatives. In the preceding there have been examples of instantaneous actions, described by the imperatives:

RESULT := RESULT + INLETS (I). FLOW
RESULT := INFLOW + PERMEATION IN + MIX SUPPLY.

These imperatives describe assignments, where the value of the quantity left to the ":=" is set to the value of the expression at the right side.

The actions of the EVAPORATION PROCESS shall define the value of EVAPORATION of the SURFACE LAYER, and this is to be done as long as the SURFACE LAYER exists as an object in the model system. As the EVAPORATION PROCESS object starts to exist at the same time as its encloser, SURFACE LAYER, and is to exist as long as the encloser exists, it shall be an unconditional time consuming action. Such an action is described by the key word "LET" followed by a property descriptor:

LET { ... }.

A property descriptor is a list of relations between the values of two quantities of the system, separated by commas and enclosed by the parentheses { and } . The effect of executing a time consuming action like this is that the relations are imposed upon the system. The fulfilment of the relations (the property) may be obtained by assigning values to the quantities of the relations. Other time consuming actions may define the values of the same quantities, and if they are executed concurrently the property imposed is the intersection of the properties.

A description of EVAPORATION PROCESS is thus:

```
SURFACE LAYER:
  WATER LAYER OBJECT BEGIN
                      .
                      .
                      .
                      EVAPORATION PROCESS:
                        OBJECT BEGIN
                          OWN TASK BEGIN
                            LET {EVAPORATION =
                                  function of the TEMPerature and the
                                  humidity of the MICRO ATMOSPHERE
                                  and of the TEMPerature of the water
                                  of the SURFACE LAYER}
                          END TASK
                        END OBJECT
                      END SURFACE LAYER OBJECT
```

The effect of executing this time consuming action is that the relation between EVAPORATION and the function at the right side is fulfilled. In this case it is obtained by changing the value of EVAPORATION and thus defining the value of this. If it is not obvious which of the values is to be changed, the description may be supplemented with "DEFINE" followed by a list of names of quantities, as e.g.

LET { ... } DEFINE EVAPORATION.

While the EVAPORATION PROCESS determines the value of EVAPORATION the values of PRECIPITATION, OUTFLOW and STREAMING are determined by the components imposing these contributions upon the SURFACE LAYER. The description of e.g. MICRO ATMOSPHERE will in this context be:

```
HYDROLOGY/HYDRAULICS LAKE :
    SYSTEM BEGIN
     :
     MICRO ATMOSPHERE :
       OBJECT BEGIN
        FUNCTION PRECIPITATION : .....;
        OWN TASK BEGIN
          LET {SURFACE LAYER. PRECIPITATION=
                   PRECIPITATION * SURFACE LAYER. AREA}
           DEFINE SURFACE LAYER. PRECIPITATION
        END TASK
       END OBJECT ;
     :
    END SYSTEM
```

1.4.2. The hydraulic Aspects

In this section the exchange of water between the layers is described.

Diffusion

One part of the exchange of water between two layers is due to diffusion, which is brought about by a combination of turbulent water movements, molecular diffusion and non-convective water movements. The turbulent diffusion will usually be the dominating. The exchange is a two-way transport of water and so that the transports in each direction are equal.

The turbulent water movements are primary due to the influence of the wind on the surface layer. From here the turbulent energy introduced by the wind is conveyed to the lower layers.

The turbulent energy of a layer is in the model represented by TURB ENERGY.

In contrast to the influence of the wind the differences in density between different layers have a stabilizing effect.

The density of water depends upon the temperature and the content of various materials. The dependence upon the temperature is the most significant; the dependence at a pressure of 1 atmosphere is shown in fig. 1.15:

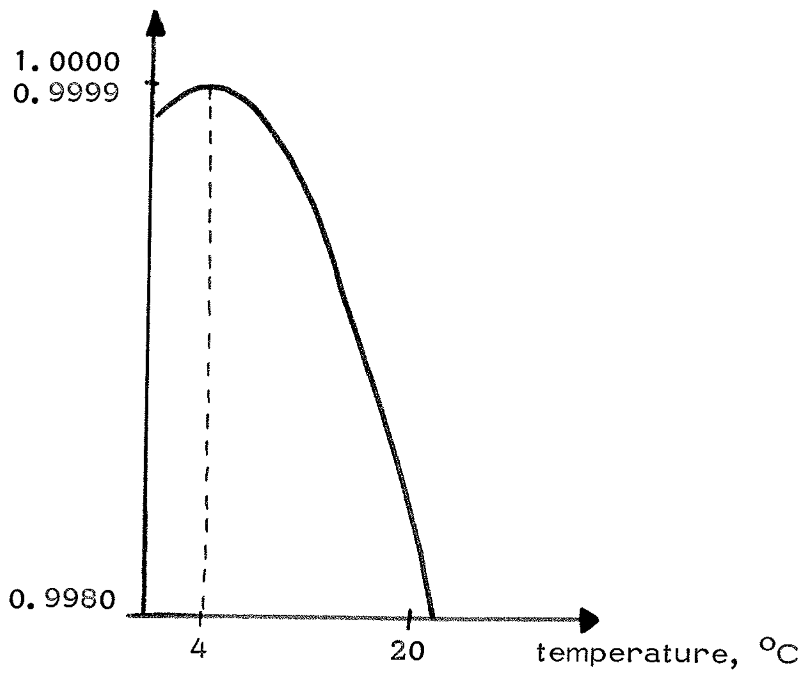


Fig. 1.15 The dependence of density of water on temperature.

As a water layer is assumed to be entirely mixed, with uniform distribution of temperature and materials, the layer will be characterized by one density and so the stabilizing energy of the layer will be zero.

As a result of the variation of solar radiation and wind (mentioned later) the temperature, and with that the density, will be different for different layers. The heavy water (at 0 - 4°C, see fig. 1.15.) will find its way to the bottom layers, while the warmer (and lighter) waters will be in the upper layers. The differences of density thus introduced between the layers will cause a stabilizing energy between layers.

The exchange of water by diffusion depends upon the stabilizing energy between the two layers. The exchange will decrease with increasing stabilizing energy. The diffusion between two layers may be described by a coefficient of diffusion. The amount of water exchanged per unit of time is proportional to the coefficient, that is

$$\text{exchanged water} = \frac{\text{diffusion coefficient} * \text{area between the two layers}}{\text{distance between the layers}}$$

The coefficient is measured in unit of area/unit of time, and the distance is measured between the centres of the layers.

A water layer will only exchange water with the layers surrounding it, so it will be characterized by a UPPER DIFF COEFF and a LOWER DIFF COEFF, both diffusion coefficients.

The water exchanged with the upper layer depends upon the surface area of the layer, while the exchange with the lower layer depends upon the undersurface area.

To describe the mixing of water by diffusion a water layer is described by:

```
CLASS WATER LAYER:
   OBJECT BEGIN
      :
      UPPER LAYER ): the layer over this layer,
      LOWER LAYER): the layer under this layer:
         REF WATER LAYER;
      FUNCTION   DENSITY, TURB ENERGY, STABIL ENERGY;
      COMMENT    DENSITY is a function of the TEMPerature,
                 TURB ENERGY of the wind and the DEPTH,
                 STABIL ENERGY of DENSITY -
                                 LOWER LAYER. DENSITY;
      SURFACE AREA, UNDERSURFACE AREA:
         QUANTITY OF AREA MEASURE;
      TYPE DIFFUSION COEFFICIENT:
         REAL MEASURE BEGIN
               measure of diffusion of water, that is of amount
               per unit of time; is measured in area/time.
            END MEASURE;
      UPPER DIFF COEFF, LOWER DIFF COEFF:
         QUANTITY OF DIFFUSION COEFFICIENT;
      :
   END OBJECT
```

The process causing diffusion between a layer and the layers surrounding it is in the model to be represented by actions, the execution of which determines the values of the exchanges with the upper and lower layers. These exchanges are represented by the quantities UPPER EXCH and LOWER EXCH, both of the type RATE OF FLOW.

Besides quantities and references an object may as attributes have action patterns defining categories of action sequences. Parts of the action sequence of an object may follow these patterns, and other objects may execute action sequences according to the patterns.

Action patterns defining sequences of instantaneous actions are in DELTA described by:

```
PROCEDURE P:
   BEGIN
      description of quantities, references and action patterns
      associated with this action entity;
      sequence of imperatives describing instantaneous actions
   END
```

"P" is the title of the pattern. Execution of the action sequence described is specified by

```
.....; P ; .....
```

Action patterns which have at least one time consuming action or which may be imposed upon another object for execution are described by

```
TASK PROCEDURE TP:
   TASK BEGIN
      :
   END TASK
```

Execution of the action sequence described is specified by

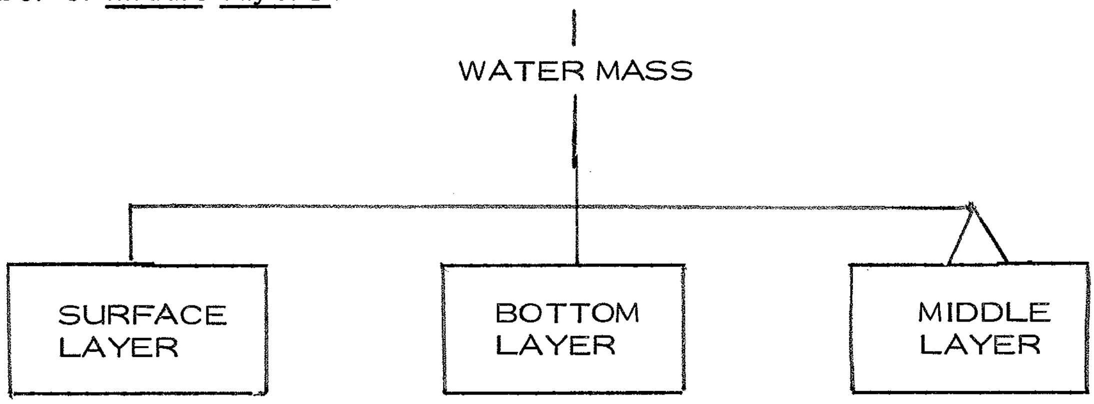
```
.....; EXECUTE TP; .....
```

A water layer's possibility of exchanging water by diffusion may then be reflected in the model by giving the WATER LAYER objects an action pattern DIFFUSION as attribute:

```
CLASS WATER LAYER:
   OBJECT BEGIN
      :
      UPPER EXCH, LOWER EXCH: QUANTITY OF DIFFUSION
                                          COEFFICIENT;
      TASK PROCEDURE DIFFUSION:
         TASK BEGIN
            LET { UPPER EXCH = (UPPER DIFF COEFF * SURFACE
                                 AREA)/(DEPTH -
                                 UPPER LAYER. DEPTH),
                  LOWER EXCH = (LOWER DIFF COEFF * UNDERSUR-
                                 FACE AREA)/
                                 LOWER LAYER. DEPTH-DEPTH)}
```

```
            DEFINE UPPER EXCH , LOWER EXCH
          END TASK;
        :
      END WATER LAYER OBJECT
```

The bottom layer does not exchange water with the sediment, nor the surface layer with the micro atmosphere, by diffusion. So it is not sufficient to divide the water mass into the surface layer and sublayers; the sublayers must further be divided into the bottom layer and a number of middle layers:



The BOTTOM LAYER and the MIDDLE LAYERs have the elements of SUBLAYER. In addition the BOTTOM LAYER and the SURFACE LAYER have as constants LOWER DIFF COEFF and UPPER DIFF COEFF respectively. The diffusion coefficients not being constants depend upon the turbulent and the stabilizing energy:

```
    CLASS WATER LAYER: ......;
    SURFACE LAYER:
      WATER LAYER OBJECT BEGIN
                     :
                     UPPER DIFF COEFF : CONSTANT
                       QUANTITY OF DIFFUSION COEFFICIENT;
                     FUNCTION LOWER DIFF COEFF;
                     depends upon TURB ENERGY and STABIL ENERGY;
                     :
                   END OBJECT;
```

```
CLASS SUBLAYER : WATER LAYER ......;
CLASS MIDDLE LAYER:
   SUBLAYER OBJECT BEGIN
                 FUNCTION  UPPER DIFF COEFF, LOWER DIFF
                 COEFF; depend upon TURB ENERGY and STABIL
                 ENERGY
             END OBJECT;

BOTTOM LAYER:
   SUBLAYER OBJECT BEGIN
                 FUNCTION UPPER DIFF COEFF;
                 depends upon TURB ENERGY and STABIL ENERGY;
                 LOWER DIFF COEFF:
                   CONSTANT QUANTITY OF DIFFUSION
                 :                         COEFFICIENT;
             END OBJECT
```

If, as here, nothing else is stated, then quantities of the type DIFFUSION COEFFICIENT will get the initial value 0.0 as it is a subtype to the type REAL. The value of a constant will be the initial value. The two constants in the preceding will thus have the value 0.0, indicating no diffusion.

Entrainment

Besides by diffusion a water layer may under certain circumstances exchange water with the lower layer by a process called entrainment. This process takes place when a thermocline between the two layers is established, that is when the density gradient is very large. The stabilizing energy between the two layers then is large, and the entrainment starts when the turbulent energy of the upper layer has become so big compared to the stabilizing energy that the water movements in the upper layer cannot be absorped by the lower layer. The turbulent energy in the upper layer causes waves between the two layers, and gradually they become so high that the wave crests tip over (see fig. 1.16).

The wave crests, consisting of water from the lower layer, are ligated and captured by the turbulent water movement in the upper layer and carried to this layer. The water elements, which by this process are carried from the upper layer to the lower layer, will because of the low turbulent energy not be captured but be carried back by bouyancy. So entrainment means a one-way transport of water from the lower to the upper layer.

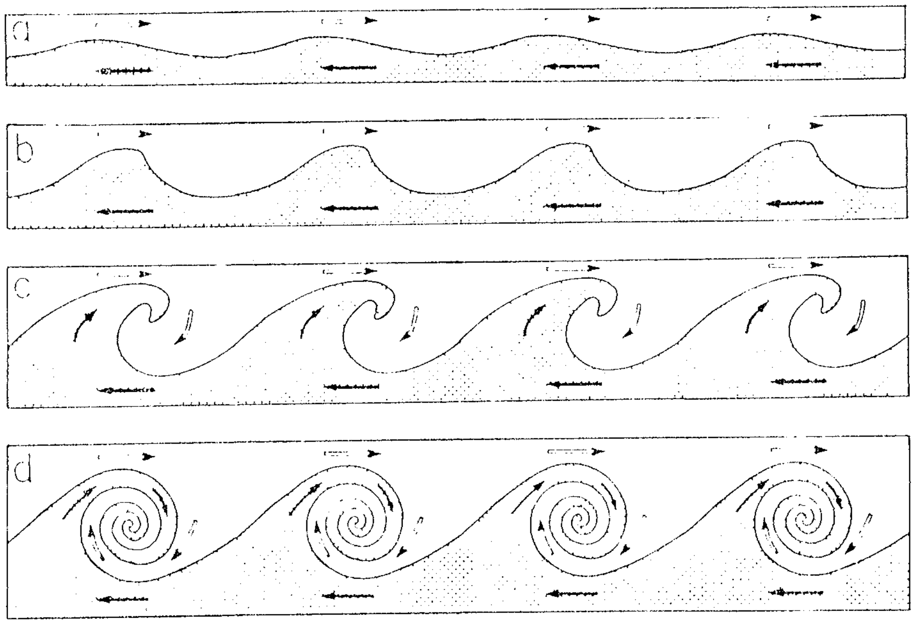


Fig. 1.16 Entrainment

As a result of the exchange of water the difference in density between the two layers decreases. The process stops when the stabilizing energy between the layers as a result of the equalization of the densities has become so small that the water movements in the upper layer can be absorped normally by the lower layer and thereby induce water movements in the lower layer.

The entrainment process is in the model represented by the action pattern ENTRAINMENT as one of the attributes of WATER LAYER:

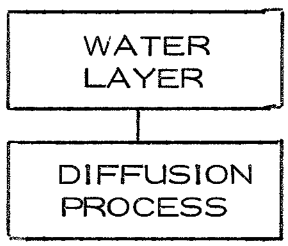
```
CLASS WATER LAYER:
   OBJECT BEGIN
      .
      .
      .
       ENT EXCH) : the amount of water exchanged with the lower
                   layer by entrainment:
          QUANTITY OF RATE OF FLOW;

       TASK PROCEDURE ENTRAINMENT:
          TASK BEGIN
             WHILE  TURB ENERGY is so big compared to STABIL
                    ENERGY that waves cannot be absorped normally
```

```
                LET  { ENT EXCH = function of
                          (TURB ENERGY - STABIL ENERGY)}
                DEFINE   ENT EXCH
           END TASK;
        :
    END OBJECT
```

Entrainment is described by use of a conditional time consuming action . As long as the assertion following "WHILE" is valid the relation of the property descriptor is imposed upon the system.

Diffusion and entrainment

While entrainment only takes place under certain circumstances there will always be a diffusion of water. This is reflected in the model by providing the WATER LAYER objects with an internal object representing the process of diffusion:



This object executes the action described in the action pattern DIFFUSION.

The process of entrainment is only performed when the turbulent energy of the water layer is big enough. Until then a water layer is regarded as passive (note that this is in the sense of hydraulics). After the entrainment (the duration is determined by the time consuming action of ENTRAINMENT) there is no water exchanged by entrainment before the next process is initiated. The value of ENT EXCH shall therefore be set to 0.0 after the entrainment. The layer is then passive again until the next process of entrainment is initiated.

So a WATER LAYER object is described to repeat as long as it exists the following sequence of actions:

- wait for entrainment to take place
- perform entrainment
- set the exchange by entrainment to 0.0

This is in DELTA described by an imperative consisting of the keyword REPEAT followed by a list of imperatives separated by "; " and enclosed by the brackets (* and *):

```
CLASS WATER LAYER:
   OBJECT BEGIN
      :
      TASK PROCEDURE DIFFUSION: .....;
      TASK PROCEDURE ENTRAINMENT: .....;
      DIFFUSION PROCESS: OBJECT BEGIN
                            OWN TASK BEGIN
                              EXECUTE DIFFUSION
                            END TASK
                         END OBJECT;
      OWN TASK BEGIN
        REPEAT (* WHILE TURB ENERGY may be absorped
                        in LOWER LAYER
                        WAIT;
                  EXECUTE ENTRAINMENT;
                  ENT EXCH := 0.0
               *)
      END TASK
   END OBJECT
```

"REPEAT" has been used before (in the description of INFLOW) but in that case the actions were only executed for I taking the values 1, 2, 3, ... NUMBER OF INLETS. Here the actions described in (* ......*) are executed repeatedly as long as the object exists in the DELTA system.

The entire supply of water by mixing is the sum of the contributions from diffusion and entrainment. The removal of water is equal to the supply:

```
CLASS WATER LAYER:
   OBJECT BEGIN
      :
      FUNCTION MIX SUPPLY:
         RATE OF FLOW BEGIN
                           RESULT: = LOWER EXCH +
                                     UPPER EXCH + ENT EXCH
                         END;
      FUNCTION MIX REMOVAL:
         RATE OF FLOW BEGIN RESULT: = MIX SUPPLY END;
      :
   END OBJECT
```

The annual temperature cycle

The lake is considered in periods of one year. The variation of the winds and of the solar radiation has the effect that the mixing and stabilizing energies alternate in being the dominating factor.

During the spring the water of the uppermost layers is warmed by the increasing solar radiation and the temperature reaches 4°C. According to the dependence between temperature and density (see fig. 1.15) the water of the upper layers will be heavier than that of the layers below, so it will sink and thus contribute to the warming of the lower layers. As the variation in density about 4°C is small (the curve of fig. 1.15 is smooth in the neighbourhood of 4°C) the stability will be almost equal to zero, so the wind is able to mix the whole water mass entirely. The lake will thus have the temperature 4°C, illustrated in fig. 1.17 with one temperature for each layer.

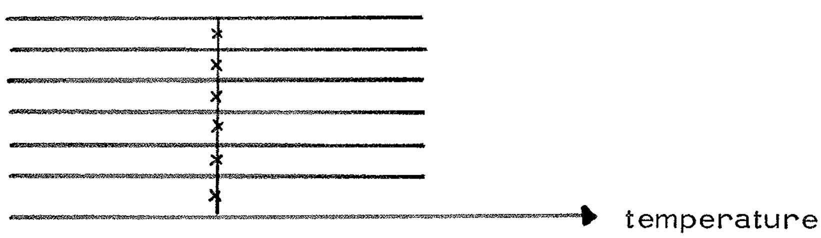


Fig. 1.17 Temperature curve

As the temperature of the upper layers increases because of the increasing solar radiation, the water of the upper layers will be lighter than that of the lower layers. This causes an increase of the stabilizing effect between the layers, and when the winds cannot defeat this stability the lake will be stratified (in what is called the "summer stagnation"). The collection of upper water layers with almost equal temperatures and higher than the temperature of the remaining layers is called the epilimnion. The zone of the sharp drop in temperature is termed the thermocline, and the collection of layers below this is called the hypolimnion (see fig. 1.18). Note that this is not a new dividing of the water mass into water layers - it is still divided in the surface layer, the middle layers and the bottom layer.

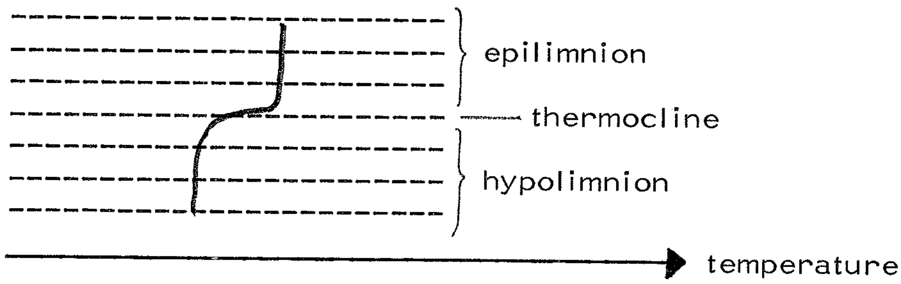


Fig. 1.18. Stratification of the water mass.

It is between the two layers at each side of the thermocline that water is exchanged by entrainment. The water exchanged causes an equalization of the temperature (and thus of the density) of the two layers, so that the thermocline after a while is "pushed" down to the zone between the next two layers (see fig 1.19).

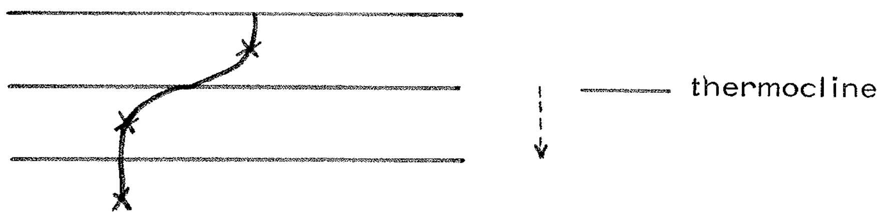


Fig. 1.19 Movement of the thermocline.

Description of the movement of the thermocline

The following model of the movement of the thermocline is valid under the assumption that the epilimnion is entirely mixed and that the diffusion coefficients of the layers of the hypolimnion decreases exponentially with decreasing depth.

The speed of the movement of the thermocline is in meters/unit of time

$$2.5 * \frac{U_F}{R_i}$$

where

$$R_i = \frac{\text{difference in density} * \text{depth of epilimnion} * \text{acc due to gravity}}{U_F^2 * \text{density of the lower layer}}$$

$$U_F = \sqrt{\frac{\text{water / air friction}}{\text{density of water}}}$$

with water/air friction = air density $*$ 10^{-3} $*$ wind velocity^2.

In the model the movement is represented by THERMOCLINE MOVEM. The value of this quantity is changed by $2.5 * U_F/R_i$ meters per unit of time, and when the value is greater than or equal to the thickness of the WATER LAYER, then this layer will no longer perform entrainment, but the next layer below.

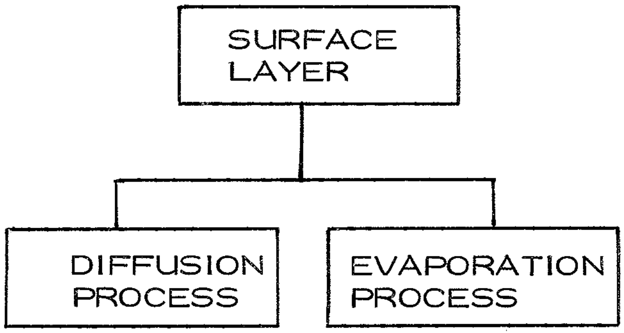
The description of the movement of the thermocline will thus be a more detailled description of the action pattern ENTRAINMENT:

```
HYDROLOGY/HYDRAULICS LAKE:
   SYSTEM BEGIN
     :
        ACC DUE TO GRAVITY : REAL CONSTANT;
        CLASS WATER LAYER:
           OBJECT BEGIN
             :
```

```
THICKNESS: QUANTITY OF LINEAR MEASURE;
TASK PROCEDURE ENTRAINMENT:
    TASK BEGIN
        FUNCTION RI:
            REAL BEGIN
                    RESULT: = (ACC DUE TO GRAVITY *
                     (LOWER LAYER. DENSITY -
                     DENSITY) * DEPTH)/(LOWER
                     LAYER. DENSITY * UF)
                END ;
        FUNCTION UF:
            REAL BEGIN
                    RESULT: = (MICRO ATMOSPHERE.
                     DENSITY * 10^-3 * MICRO ATMOS-
                     PHERE. WIND VELOCITY^2/
                     DENSITY)^-1/2
                END;
        THERMOCLINE MOVEM: QUANTITY OF LINEAR
                                          MEASURE;
        WHILE THERMOCLINE MOVEM <
                          LOWER LAYER. THICKNESS
            LET {change of THERMOCLINE MOVEM per
                 unit of time = 2.5 * UF/RI;
                 ENT EXCH = the amount of water ex-
                            changed per unit of time
                            to move the thermocline
                            the distance 2.5 * UF/RI
                            per unit of time}
            DEFINE THERMOCLINE MOVEM, ENT EXCH
    END ENTRAINMENT TASK;
  :
 END WATER LAYER OBJECT ;
:
END SYSTEM
```

Final remarks on the model

In the preceding the WATER MASS has been split into the singular objects SURFACE LAYER and BOTTOM LAYER and a number of MIDDLE LAYERs. Every WATER LAYER object contains a DIFFUSION PROCESS object and has as an attribute the action pattern ENTRAINMENT. Furthermore, the SURFACE LAYER contains an EVAPORATION PROCESS object, so that this object is illustrated:



In the SURFACE LAYER there will always be an EVAPORATION and a DIFFUSION, while in the other layers there will only be a DIFFUSION.

The model system has thus been described by the singular objects and the classes of objects contained in the system, but nothing has been said about what goes on in the model system itself.

When a system generator on basis of the DELTA system description generates a model system it is done by generating a DELTA system object, this object representing the referent system as a whole.

At the moment of generation of the system object the singular objects contained in it are generated, and they all (also the system object) start at the same moment to execute the actions of their OWN TASKs.

But, descriptions of classes of objects, e.g. INLET and MIDDLE LAYER, do not introduce any objects of the classes into the model system. When generating HYDROLOGY/HYDRAULICS LAKE as a model of a lake, as e.g. Lake Esrom, then it has to contain a certain number of objects of these classes to represent the corresponding components of the lake.

As a whole the model system is characterized by number of inlets and number of middle layers, and they shall at the generation get the values reflecting the referent system. This is described by:

```
HYDROLOGY/HYDRAULICS LAKE:
   SYSTEM BEGIN
         :
         NUMBER OF INLETS, NUMBER OF MIDDLE LAYERS:
            INTEGER CONSTANT;
         :
   END SYSTEM
         PUT ( * NUMBER OF INLETS: = .....;
                 NUMBER OF MIDDLE LAYERS: = .....; *)
```

After these initializations the system object starts the execution of its OWN TASK. One of the first actions shall be to generate the INLET and MIDDLE LAYER objects. When they are generated they will start the execution of their action sequence.

The system object is not characterized by other actions - as long as the model system is considered the dynamics of the model are represented by the internal objects' execution of actions:

```
HYDROLOGY/HYDRAULICS LAKE:
   SYSTEM BEGIN
         :
         OWN TASK BEGIN
            generate NUMBER OF INLETS objects of the class
            INLET and NUMBER OF MIDDLE LAYERS objects
            of the class MIDDLE LAYER according to the charac-
            teristics of the corresponding components of the refe-
            rent system;
            WHILE TIME < time of consideration WAIT;
         END TASK
   END SYSTEM
         PUT (* ....... *)
```

To generate objects according to the characteristics of the referent system means that quantities, as e.g. DEPTH, THICKNESS, UNDER SURFACE, and SURFACE of MIDDLE LAYER objects are assigned values corresponding to the middle layers, which the objects represent.

The generation of an object is described by an imperative consisting of the key word NEW followed by a class title:

```
NEW MIDDLE LAYER  PUT ( *DEPTH : = .....;
                         THICKNESS : = .....;
                            :
                                              *)
```

To denote INLET objects there are introduced the references INLET (1), INLET (2), ..., INLET (NUMBER OF INLETS). So the generation of an INLET object has the form

```
INLET (I) : - NEW INLET  PUT (* ······*)
```

The value of this reference is the generated object.

Quantities of the singular objects of the model system shall also have values according to the referent system. As these objects are generated as part of the generation of the system object, the descriptions of the singular objects may be supplemented with specifications of initializations. Thus, the SURFACE LAYER is to be generated with initializations of the same quantities as of the MIDDLE LAYERs:

```
HYDROLOGY/HYDRAULICS LAKE:
  SYSTEM BEGIN
     :
     SURFACE LAYER:
        OBJECT BEGIN ..... END OBJECT
          PUT ( * DEPTH : = .....;
                  THICKNESS : = .....;
                      :
                                        *)
     :
  END SYSTEM
```

1.5 Biological Description of a Lake

A description of the biological aspects of a lake is a description of the organic and inorganic matters in the water and the sediment, and a description of the biological processes changing the content and the composition of these matters.

With respect to eutrophication the water of a lake is characterized by its content of algae plankton, zooplankton, detritus, fish, inorganic phosphorus and nitrogen, organic phosphorus and nitrogen (in algae and detritus) and dissolved oxygen. The primary production of organic matter requires many other elements (e.g. metals), but they are never reducing factors as e.g. phosphorus and nitrogen.

The starting point in developing a biological model of a lake is a division of the water mass into layers, which may be regarded as entirely mixed, that is they may be characterized by the biomasses of the organic matters and by the concentrations of the inorganic matters. Thus, there is not distinguished between different species,of e.g. algae.

A water layer will in the model system BIOLOGY LAKE be represented by a WATER LAYER object. As the water mass will be divided into several layers the common features of the layers will be described in a class of WATER LAYER objects.

A first description of WATER LAYER is:

```
CLASS WATER LAYER:
    OBJECT BEGIN
        TYPE BIO MASS : REAL MEASURE BEGIN
                                            measure for content of organic
                                            matters, in gram/m3
                                         END MEASURE;
        TYPE CONCENTRATION: REAL MEASURE BEGIN
                                                   measure for content of
                                                   inorganic matters, in gram/m3
                                                END MEASURE;

        INORG P, INORG N,  DP ) : detritus phosphorus,
        DN ) : detritus nitrogen,  OXYGEN : QUANTITY OF CONCENTRA-
                                                                TION;
```

AP) : algae plankton, ZP) : zooplankton, D) : detritus,
FISH : QUANTITY OF BIO MASS;

⋮

END OBJECT

Furthermore, a water layer is characterized by some processes changing its content of different matters. These processes are photosynthesis, zooplankton's consumption of algae, fish's consumption of zooplankton, inactivation of plankton (by death), respiration (of all organisms), chemical and bacterial decomposition (of detritus to nutritive salts) and sedimentation (of algae and detritus).

The values of the variables INORG P, INORG N, AP, ZP, ... define the state of a water layer and are therefore called state variables. The value of these are changed a.o. by the processes mentioned above. So besides the processes a water layer is characterized by a mass balance for each of the state variables. These mass balances characterize the water layer as a whole, as they express the dependencies between state variables.

A mass balance is expressed by defining the value of a state variable by its change per unit of time, and this change is expressed by a function of contributions from different processes. The single contribution is of a type biomass change (or concentration change) which may be defined as a subtype to biomass (concentration). The contribution is besides e.g. the biomass characterized by a unit of time and by operations covering conversions between different units:

```
TYPE BIOMASS CHANGE :
        BIOMASS MEASURE BEGIN .... END MEASURE.

TYPE CONCENTRATION CHANGE:
        CONCENTRATION MEASURE BEGIN .... END MEASURE.
```

Considering e.g. AP (algae plankton) the changes of its value are due to:

netto production by photosynthesis, decrease by consumption, decrease by inactivation (biological changes) and abiological changes, as e.g. sedimentation of algae (that is from a water layer to the layer below).

The description of a water layer's content of algae plankton thus consists of a description of the contributions to the change of the content and a description of the mass balance:

```
CLASS WATER LAYER:
  OBJECT BEGIN
      .
      .
      .
      PRODUCTION, AP CONSUM, AP INACT,
      AP SEDIMENT): algae plankton sedimented from this layer,
      SEDIMENT AP): sedimented algae plankton to this layer, ...:
         QUANTITY OF BIOMASS CHANGE;

      OWN TASK BEGIN
         LET { change of AP/unit of time =
                    PRODUCTION - AP CONSUM
                    + SEDIMENT AP - AP SEDIMENT + ... }
            DEFINE AP
      END TASK
  END WATER LAYER OBJECT
```

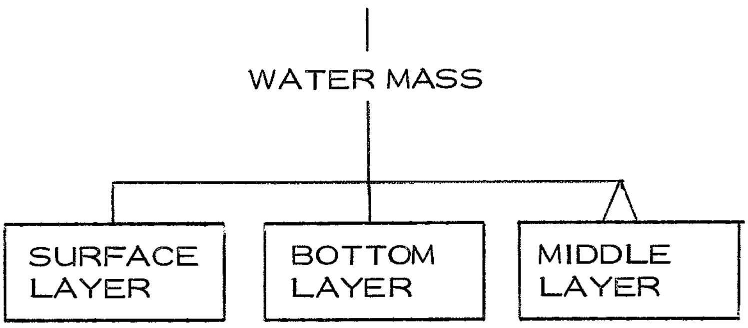
The values of the contributions are determined by the corresponding processes, that is PRODUCTION by the photosynthesis process, SEDIMENT AP by the sedimentation process etc....

Correspondingly the mass balances of the other state variables are described. A further biological description will be a description of the various processes. In the following only the exchange of matters between the water mass and the sediment will be described.

Exchange between water mass and sediment

In order to describe the exchange of materials between the water mass and its environment it is necessary to divide the water in a number of layers. The rainfall and the outlet only influence the surface layer of the lake, and when considering the exchange between the water and the sediment the lower part of the water mass must be divided into layers. The exchange depends upon the concentrations in the water, so it would be too simplified to operate with the mean values of the hypolimnion. The dividing into layers must take place at a time with great concentration gradients, so that the property of each layer being entirely mixed may be obtained. Each water layer will then be characterized by the mean value of the concentrations in the layer.

The water mass is divided into a surface layer, bottom layer and a number of middle layers. These components are in the model represented by:

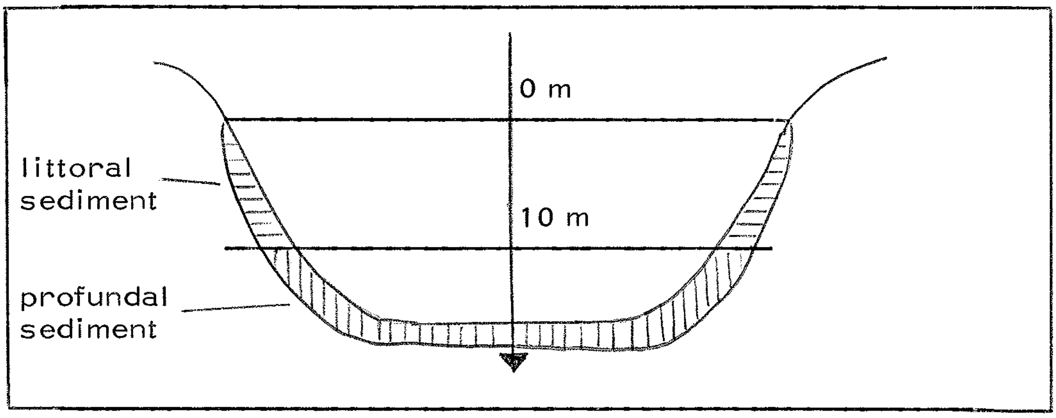


The argument for splitting the WATER MASS is that the single components exchange directly with the components of the environment. Even though these layers are different they have common features described in the class WATER LAYER.

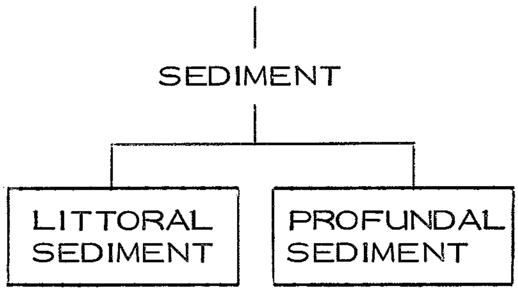
The sediment is able to take up phosphorus and act as an "ecological buffer" that is phosphorus is released from the sediment when the input of phosphorus is diminished. The detritus sedimented from the water is decomposed and the resulting nutritive salts may be released from the sediment.

Sedimentation

From a water layer detritus is sedimented to the water layer below and to the part of the sediment with which it is in contact. The sediment might be divided so that each water layer had its sediment layer. But it is sufficient to divide the sediment into the part over the depth of 10 meters and the part below the 10 meters:



As the water layers exchange directly with the sediments and as the sediment is entirely described by the two parts, the SEDIMENT of the model is split:



The two objects have features in common as sediments:

```
BIOLOGY LAKE:
    SYSTEM BEGIN
        .
        .
        .
        CLASS SEDIMENT: OBJECT BEGIN
                                common features of sediments
                              END OBJECT;
```

```
    LITTORAL SEDIMENT:
       SEDIMENT OBJECT BEGIN
                    the part of the sediment over the depth
                    of 10 meters
                  END OBJECT;
    PROFUNDAL SEDIMENT:
       SEDIMENT OBJECT BEGIN
                    the part of the sediment below the
                    depth of 10 meters
                  END OBJECT;
       :
 END SYSTEM
```

For the purpose of describing the exchange of phosphorus between sediments and water layers the sediments are described by:

```
CLASS SEDIMENT :
   OBJECT BEGIN
      D): detritus : QUANTITY OF BIOMASS;
      INORG P: QUANTITY OF CONCENTRATION;
      SEDIMENT D): sedimented detritus from the layers in
                   contact with this sediment,
      DECOM D ): decomposed detritus:
         QUANTITY OF BIOMASS CHANGE;
      OWN TASK BEGIN
         LET { change of D/unit of time =
                  SEDIMENT D - DECOM D
                  + inactivated algae }
           DEFINE D
      END TASK
   END OBJECT
```

The value of SEDIMENT D depends upon the water layers in contact with the sediment, the speed of sedimentation and the size of the surface of contact. The value of SEDIMENT D is the sum of the sedimentations from the water layersin contact with the sediment.

In the following description of the water/sediment phosphorus exchange is included only one sediment and the water layers in contact with it (see fig. 1. 20).

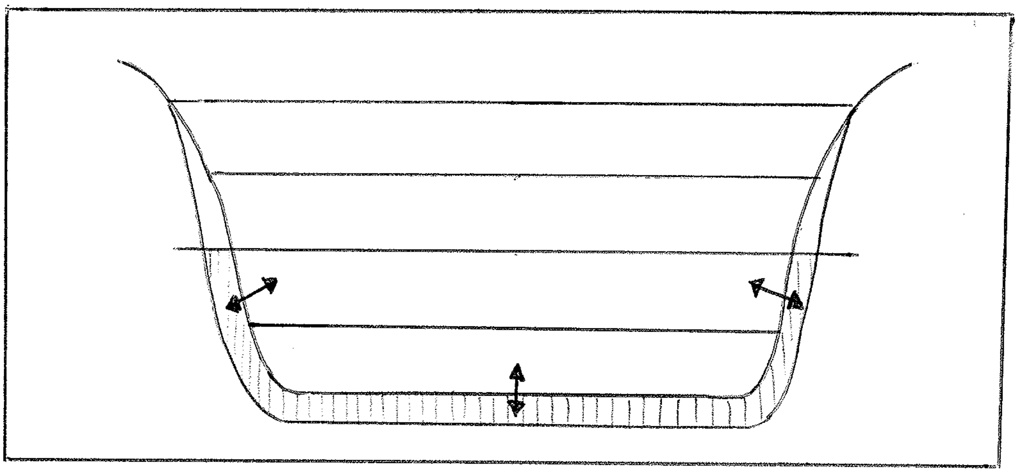


Fig. 1.20

The phosphorus released from the sediment is a result of a decomposition of the sedimented detritus. The phosphorus released contributes to the change of the inorganic phosphorus concentrations in the water layers. The contributions are in the single WATER LAYER represented by REL P.

The release of phosphorus is a result of three processes:

- a biological decomposition of detritus on the surface of the sediment; the contribution to REL P is called BIO REL P
- a desorption ; the contribution called D REL P
- a diffusion, starting when the bottom water becomes anaerob; the contribution called DIFF REL P.

The value of the release REL P is the sum of the values of BIO REL P, D REL P and DIFF REL P.

The three processes will in the SEDIMENT be represented by BIO DECOMPOSITION, DESORPTION and DIFFUSION.

Biological decomposition

BIO DECOMPOSITION describes a decomposition (mineralization) of detritus on the surface of the sediment. The detritus decomposed is detritus contained in the water and not detritus of the sediment. SEDIMENT D thus represents the detritus reaching the SEDIMENT without being decomposed (see fig. 1.21).

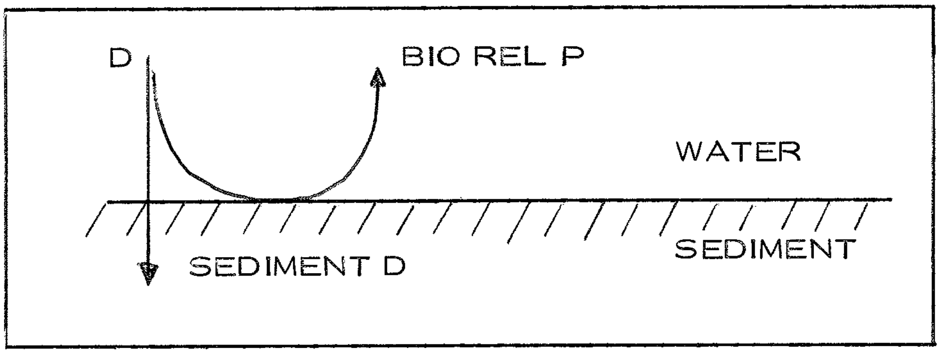


Fig. 1.21. Biological decomposition

The decomposition depends upon the kind of the sediment (represented by BIO CONST) and the temperature of the water. The amount of detritus decomposed depends upon the biomass of the detritus - this dependency is described by MIN CONST:

```
WL : ARRAY (1: number of water layers in contact with
                 this sediment) OF REF WATER LAYER;
TASK PROCEDURE BIO DECOMPOSITION:
  TASK BEGIN
    LET { WL(k). BIO REL P =
            BIO CONST * EXP (0.20 * WL(k).TEMP)
              for k = 1,2,.....,number of water layers,
            WL(k).DECOM D = MIN CONST * WL(k).D
              for k = 1,2,.....,number of water layers }
        DEFINE WL.BIO REL P, WL.DECOM D
  END TASK
```

Desorption

The phosphorus which reach the sediment with the sedimented detritus is divided into netto sedimented phosphorus and phosphorus which after a decomposition (of detritus) in the sediment contributes to its content of inorganic phosphorus in the interstitial water, from where it may be released to the water by desorption and diffusion (see fig. 1.22).

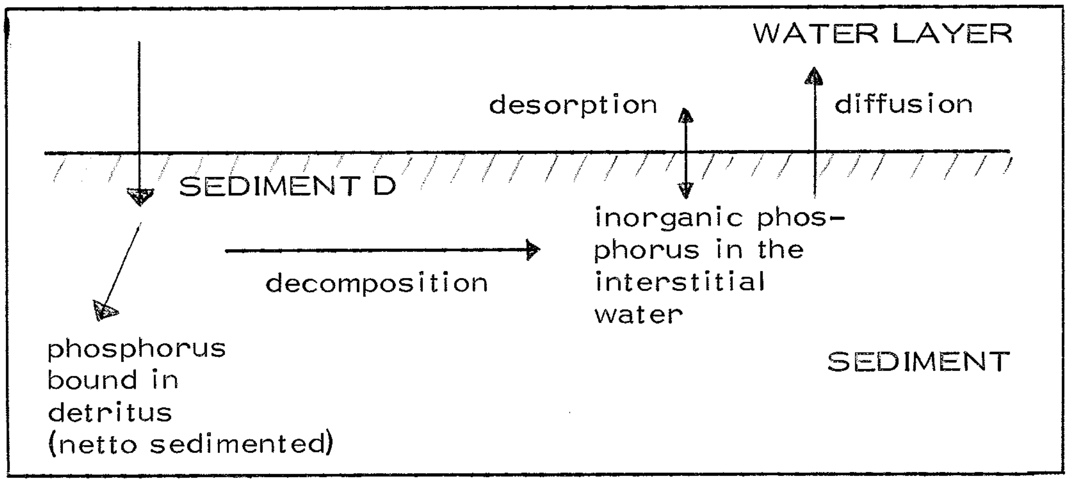


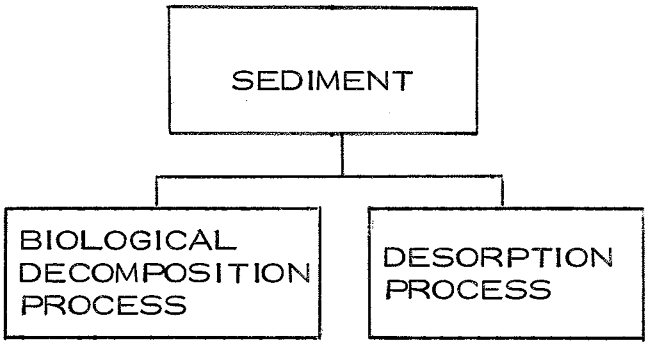
Fig. 1.22 Desorption and diffusion

The desorption is a gradient dependent process:

```
TASK PROCEDURE DESORPTION:
  TASK BEGIN
    LET { WL(k).D REL P =
            0.6 * ln (WL(k).INORG P) - 2.27
             for k = 1, 2, ....., number of water layers,
          DESORPED P = sum  of WL(k).D REL P, k=1,2,...
      DEFINE WL.D REL P, DESORPED P
  END TASK
```

DESCORPED P represents the total phosphorus released from the sediment by desorption.

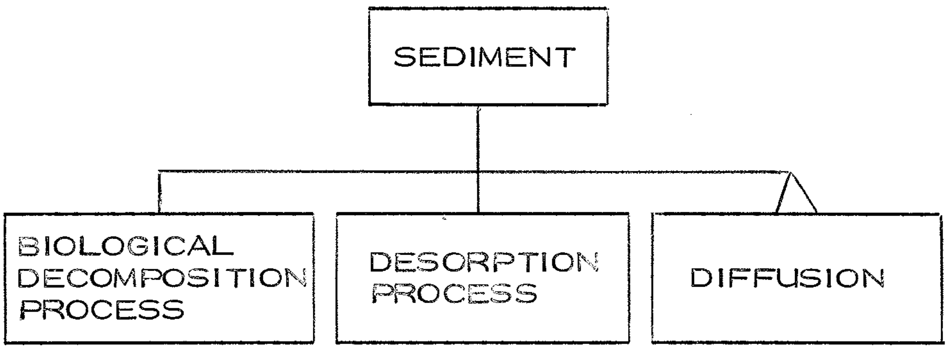
The biological decomposition and the desorption has been described by task procedures which the SEDIMENT objects may execute. Now, the sediment will release phosphorus concurrently by decomposition and desorption and they will take place all the time. This is reflected in the model by having a SEDIMENT object contain two objects:



each having as its own task the execution of the corresponding task procedure.

Diffusion

Release by diffusion does not take place all the time. It starts when the water layer in contact with the sediment becomes anaerob. So if there are more than one anaerob water layer there will be more than one diffusion process in action. The diffusion is therefore not described by a task procedure but by a class of DIFFUSION objects contained in the SEDIMENT objects:



Each object of this class represents a diffusion process.

A diffusion process defines the value of the contribution DIFF REL P of the WATER LAYER which has initiated the process, and this is valid as long as the WATER LAYER is anaerob. This is described by a conditional time consuming action:

```
WHILE WATER LAYER. OXYGEN CONDITION = ANAEROB
   LET { WATER LAYER. DIFF REL P = .....} DEFINE
                                    WATER LAYER. DIFF REL P
```

The value of DIFF REL P depends upon the phosphorus concentration in the WATER LAYER and the phosphate concentration in the interstitial water, PHOSPH INT. By the diffusion this concentration will be diminished by DIFF PHOSPH INT:

```
CLASS SEDIMENT:
   OBJECT BEGIN
      .
      .
      .
      PHOSPH INT : QUANTITY OF CONCENTRATION;
      CLASS DIFFUSION:
         OBJECT BEGIN
            WL ) : the water layer which has initiated
            this process : REF WATER LAYER;

            OWN TASK BEGIN
               WHILE WL. OXYGEN CONDITION = ANAEROB
                 LET { WL. DIFF REL P =
                         1.21 * (PHOSPH INT - WL. INORG P) - 1.70,
                       DIFF PHOSPH INT = WL. DIFF REL P }
                   DEFINE WL. DIFF REL P, DIFF PHOSPH INT
            END TASK
         END DIFFUSION OBJECT ;
      .
      .
      .
   END SEDIMENT OBJECT
```

When a water layer becomes anaerob a diffusion to the layer is started. In the model this is described by having the WATER LAYER object cause the corresponding SEDIMENT object to start a diffusion process, that is to generate an object of the class DIFFUSION. While the water layer is aerob and while the diffusion takes place the layer is described as passive, and after the completion of the diffusion the contribution to the phosphorus of the water layer (represented by DIFF REL P) shall be set to zero:

```
CLASS WATER LAYER:
   OBJECT BEGIN
      :
      SED ): the sediment in contact with this
               layer : REF SEDIMENT;
      OXYGEN : QUANTITY OF CONCENTRATION;
      OXYGEN CONDITION :
         MEASURE BEGIN
            measure of the oxygen content;
            may have the values AEROB and ANAEROB;
            if OXYGEN ≤ 1 mg/l then ANAEROB else AEROB
         END MEASURE;
      OWN TASK BEGIN
         REPEAT
         (* WHILE OXYGEN CONDITION = AEROB WAIT;
            INTERRUPT SED BY SED. START DIFFUSION
               PUT (* WATER LAYER := THIS WATER LAYER *)
            WHILE OXYGEN CONDITION = ANAEROB WAIT;
            DIFF REL P : = 0
          *)
      END TASK
   END OBJECT
```

The task of starting a diffusion is in SEDIMENT described by the task procedure START DIFFUSION. In order to start a diffusion it must be known to which water layer it shall contribute. This information to the DIFFUSION object is communicated by use of the reference WATER LAYER of START DIFFUSION. When a WATER LAYER initiates a diffusion the reference WATER LAYER of START DIFFUSION is set to the actual WATER LAYER (denoted by "THIS WATER LAYER"). The description of this initiation has been:

```
INTERRUPT SED BY SED. START DIFFUSION
   PUT (* WATER LAYER : - THIS WATER LAYER *)
```

that is, the WATER LAYER imposes the execution of START DIFFUSION upon the SEDIMENT object denoted by SED.

START DIFFUSION generates an object of the class DIFFUSION. The WATER LAYER object to which the DIFFUSION object is to contribute is denoted by the reference WATER LAYER of START DIFFUSION, so in the generation of the DIFFUSION object the reference WL of the DIFFUSION object is set to the value of WATER LAYER:

```
CLASS SEDIMENT:
  OBJECT BEGIN
      .
      .
      .
      TASK PROCEDURE START DIFFUSION:
        TASK BEGIN
          WATER LAYER : REF WATER LAYER;
          NEW DIFFUSION PUT (* WL : - WATER LAYER *)
        END TASK;
      .
      .
      .
  END OBJECT
```

The execution of the action described by the imperative "NEW DIFFUSION ....." brings into existence a DIFFUSION object, and it will immediately start the execution of its own task, thus defining the value of DIFF REL P (as long as the WATER LAYER is ANAEROB). At a given moment several different DIFFUSION objects may exist, each contributing to a WATER LAYER.

As for the HYDROLOGY/HYDRAULICS LAKE the own task of the model system BIOLOGY LAKE will be generation of WATER LAYER objects followed by an action defining the system object as passive:

```
BIOLOGY LAKE:
  SYSTEM BEGIN
      .
      .
      .
      NUMBER OF WATER LAYERS : INTEGER;
      OWN TASK BEGIN
        generate NUMBER OF WATER LAYERS objects of the
        class WATER LAYER in accordance with the corre-
```

```
            sponding components of the referent system;
            WHILE TIME < time of consideration WAIT
        END TASK
    END SYSTEM
                PUT (* NUMBER OF WATER LAYERS : = ..... *)
```

The singular objects LITORAL SEDIMENT and PROFUNDAL SEDIMENT of the system are generated at the same time as the system object. They have no characteristics in addition to those given in the class SEDIMENT, but quantities have different values reflecting that they represent two kinds of sediment. Especially the BIO CONSTs have different values, reflecting that the biological decomposition depends upon the nature of the sediment.

```
BIOLOGY LAKE:
    SYSTEM BEGIN
        :
        CLASS SEDIMENT : OBJECT BEGIN ..... END OBJECT;
        LITTORAL SEDIMENT:
            SEDIMENT OBJECT PUT (* BIO CONST : = 0.19;.....*);
        PROFUNDAL SEDIMENT:
            SEDIMENT OBJECT PUT (* BIO CONST : = 0.56;.....*);
        :
    END SYSTEM
```

1.6. Mathematical Description of a Lake

The task of the model group of FSP has been to formulate a mathematical lake model on the basis of the work of the other groups, and this model is made operational by being transformed into a DYNAMO model, DYNAMO being a special simulation language.

In the following the DELTA language will be used to describe parts of the mathematical model and the same parts of the DYNAMO model. The first use of the context concept and a distinction between context and system class are demonstrated.

1.6.1. Mathematical Lake Model

The mathematical model is worked out using the concepts of System Dynamics. To indicate this in the description of the model the title "System Dynamics" may be used as prefix:

MATHEMATICAL LAKE:
SYSTEM DYNAMICS SYSTEM BEGIN ..... END SYSTEM

This has been done before to indicate the context in which the model is described (when collecting the concepts LINEAR MEASURE, AREA MEASURE and VOLUME MEASURE for use in the hydrological/hydraulic model, see p.).

The context System Dynamics

For some people the title "System Dynamics" will be sufficient: the contexts in which they generate model systems by interpreting system descriptions contain the System Dynamics concepts, or they know the concepts so well that they immediately "understand" the system description.

In most cases it will be necessary with a description of the context. As a first solution I suppose that a context is described by a text enclosed by the parantheses CONTEXT BEGIN and END CONTEXT, preceded by the name of the context, as

SYSTEM DYNAMICS : CONTEXT BEGIN ..... END CONTEXT.

This context description together with the system description should enable a system generator to interpret the system description in a context containing the concepts of System Dynamics.

The following description of the context SYSTEM DYNAMICS includes only the concepts being necessary for the description of the mathematical model.

SYSTEM DYNAMICS:

CONTEXT BEGIN

A system is characterized by a number of level variables (state variables), the values of which at a given moment define the state of the system, and by a number of rates (process variables) describing the dynamics of the system as their values define the rates at which the values of the levels change.

The relation between a level and a rate is expressed by a differential equation

$$\frac{d\ L(t)}{dt} = R(t)$$

where L is a level and R is a rate. d L(t) denotes the change of the value of L. The differential equation expresses that the value of L is changed by $R(t) * dt$ in the time dt. Putting $t' = t - dt$ and having L(t) denoting the value of L at the time t, then

$$L(t) = L(t') + R(t) * dt$$

The changes of the levels may depend upon time, that is the rates depend upon time. This is expressed by having R depend upon t. If the change depends upon other levels, then this is expressed by defining R as a function of levels.

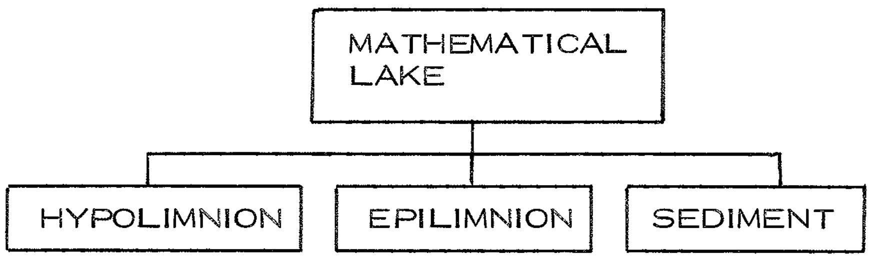
Feedback in a system is described by having R(t) depend upon the level changed by R(t).

If a system is characterized by more than one relation between levels and rates, then the dynamics of the system is described by a number of differential equations and rate functions.

END CONTEXT

The mathematical lake model

The basis of the mathematical lake model is the decomposition into the components epilimnion, the upper part of the water mass with no temperature gradients, hypolimnion, the part below the epilimnion, and the sediment. These are in the model system represented by the objects:



Note that the mathematical model does not contain all the components identified in section 1.2, but only the water mass and the sediment. The other components are implicitly contained in the model as they are represented by functions affecting the levels of the three components without being affected themselves (e.g. solar radiation and rainfall).

The three components are characterized by a number of level variables essential to the energy conversion of the system and by a number of rate variables describing biological and abiological processes.

HYPOLIMNION and EPILIMNION are both water layers, so they have some common characteristics, represented by the class WATER LAYER. The two layers are assumed to be entirely mixed, so their contents are indicated by the biomasses of organic matter and the concentrations of inorganic matter, and so is the SEDIMENT:

```
MATHEMATICAL LAKE:
  SYSTEM DYNAMICS
    SYSTEM BEGIN
      CLASS WATER LAYER:
        OBJECT BEGIN
          CP): conc of inorganic phosphorns, CN): conc
          of inorganic nitrogen, DP): conc of detritus
          phosphorus, DN): conc of detritus nitrogen,
          AC): biomass of plankton algae, ZC): biomass
          of zooplankton, FISK): fish biomass, OO): conc
          of oxygen: QUANTITY OF LEVEL;
          :
        END OBJECT;
      EPILIMNION: WATER LAYER OBJECT;
      HYPOLIMNION: WATER LAYER OBJECT;
      SEDIMENT: OBJECT BEGIN
                  CP, CN, DP, DN, AC: QUANTITY OF LEVEL;
                  :
                END OBJECT
    END SYSTEM
```

The names of the variables are the same as used by the model group.

The processes of the lake change the content of organic and inorganic matter. In the mathematical model this is described by the changing of the values of the level variables.

The value of each level variable is changed as a result of biological and abiological processes. This is mathematically expressed by a differential equation:

$$\frac{d\ \text{level}}{dt} = \text{change as result of biological processes} + \text{change as result of abiological processes}$$

These changes are represented by rate variables. For each level variable, e.g. AC, FAC represents the biological change and AAC the abiological change. So a WATER LAYER is further described by:

```
CLASS WATER LAYER:
   OBJECT BEGIN
        :
        FCP, FCN, FDP, FDN, FAC, FZC, FFISK, FOO,
        ACP, ACN, ADP, ADN, AAC, AZC, AOO:
           QUANTITY OF RATE;
        :
   END OBJECT
```

Furthermore, a WATER LAYER is characterized by a mass balance for every level variable:

```
CLASS WATER LAYER:
   OBJECT BEGIN
        :
        OWN TASK BEGIN
           LET { dAC/dt = FAC + AAC,
                 dZC/dt = FZC + AZC,
                          :
                                      }
             DEFINE AC, ZC, ....
        END TASK
   END OBJECT
```

Each of the changes (FAC, AAC, ...) is a result of more than one process. The biological change of ZC is thus equal to the difference between growth by consumption of algae and decrease by death. Correspondingly the value of FAC is production by photosynthesis – algae consumed by zooplankton – decrease by death.

The rate variables are thus functions of level variables, the values of which are determined by the different processes of the layer:

```
WATER LAYER:
   OBJECT BEGIN
        :
```

```
RPROD): algae production by photosynthesis,
ZAC): algae consumed by zooplankton,
DAC): decrease of algae
    QUANTITY OF LEVEL ;
FUNCTION FAC:
   RATE BEGIN
          RESULT: = RPROD - ZAC - DAC
       END;
VAC): growth of zooplankton by consumption of algae,
DZC): decrease of zooplankton:
    QUANTITY OF LEVEL;
FUNCTION FZC:
   RATE BEGIN
          RESULT: = VAC - DZC
       END ;
  .
  .
  .
END OBJECT
```

Both layers are characterized by the processes <u>herbivor consumption</u>, carnivor consumption, chemical and bacterial decomposition and <u>inactivation and metabolism</u>. Mathematically these processes are described in the same way in the two layers, so they are defined in the class WATER LAYER. Herbivor consumption is described by defining the values of the rates VAC and ZAC, while the values of DZC and DAC are defined in inactivation and metabolism:

```
WATER LAYER:
   OBJECT BEGIN
      .
      .
      .
      TASK PROCEDURE HERB CONS:
         TASK BEGIN
            VEF): theoretical growth rate of zooplankton,
            KG): effect of zooplankton's consumption at actual tempe-
            rature, DECON): zooplankton's speed of consumption
            of detritus and bacteria  :QUANTITY OF RATE;
```

```
            LET { VAC = VEF * (KG + 0.1 * DECON),
                      ZAC = -KG * ZC }
              DEFINE VAC, ZAC
          END TASK;

        TASK PROCEDURE INACT AND METABOL:
          TASK BEGIN
            KDZC): speed of decrease by death of zooplankton:
              QUANTITY OF RATE;
            KDAC): speed of decrease by death of algae plankton:
              CONSTANT QUANTITY OF RATE;
            LET { DZC = KDZC * ZC,
                      DAC = KDAC * AC }
              DEFINE DZC, DAC
          END TASK;
      .
      .
      .
      END OBJECT
```

In a more detailed description of WATER LAYER VEF, KG and DECON will be defined as functions of AC, ZC and the temperature of the water. The functions are the same for the two layers, so they will be attributes of WATER LAYER.

One of the differences between epilimnion and hypolimnion is that in the epilimnion there is a production of algae by photosynthesis and that there is no photosynthesis process in the hypolimnion. This is reflected by redefining RPROD of HYPOLIMNION to a constant, having the initial value 0, and by letting the process PHOTOSYNTHESIS of EPILIMNION define the value of RPROD:

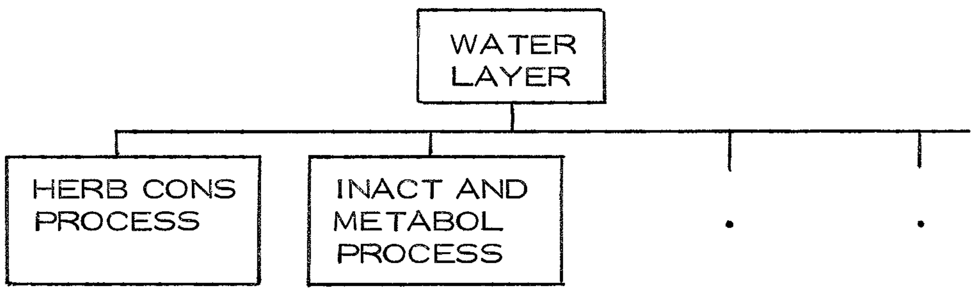
```
  HYPOLIMNION:
    WATER LAYER OBJECT BEGIN
                        RPROD: CONSTANT QUANTITY OF LEVEL;
                     END OBJECT
                        PUT (* RPROD : = 0 *)
```

and

```
EPILIMNION:
    WATER LAYER OBJECT BEGIN
                         TASK PROCEDURE PHOTOSYNTHESIS:
                           TASK BEGIN
                             define the value of RPROD as a function
                             of number of hours of sunshine of a day,
                             the light intensity and the concentra-
                             tions of inorganic phosphorus and nitrogen.
                           END TASK;
                             :
                        END OBJECT
```

Until now a water layer is only described by the levels, rates and processes characterizing it. It is not stated when these processes take place, whether it happens in a series or not.

All processes of the water layers take place concurrently and they take place at the same time as a mass balance is kept for every level, that is at the same time as the execution of the own tasks of the WATER LAYERs. A WATER LAYER thus contains a number of objects, representing the processes:



Each of the objects have as their own task the execution of the corresponding action pattern, that is

HERB CONS PROCESS : EXECUTE HERB CONS

and

INACT AND METABOL PROCESS : EXECUTE INACT AND METABOL.

Besides these objects the EPILIMNION contains an object representing the photosynthesis process and thus executing PHOTOSYNTHESIS as its own task.

In the preceding only the level variables AC and ZC and the biological processes affecting these are treated. The other variables are treated in the same way.

1.6.2. DYNAMO Model of a Lake

To solve the set of differential equations, which the mathematical model consists of, the model group has used the simulation language DYNAMO, which is based upon the concepts of System Dynamics. A DYNAMO system is characterized by a number of level variables and rate variables and by a set of equations, which - like the differential equations of the mathematical model - defines the relation between the various level and rate variables. A DYNAMO system consists only of one component (the system itself) and may not contain internal components representing internal processes.

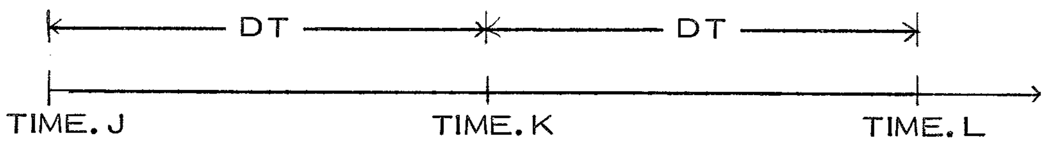
To describe the DYNAMO model I will describe a pattern defining DYNAMO systems, that is a description of the common properties of DYNAMO models. This pattern is a class of DELTA model systems, with the title "DYNAMO", and it may be used in description of concrete DYNAMO models, as e.g.:

DYNAMO LAKE MODEL: DYNAMO SYSTEM BEGIN.....END SYSTEM

In the system descriptor SYSTEM BEGIN ..... END SYSTEM the elements of the specific DYNAMO system are described.

In a DYNAMO system the time is divided into intervals of the length DT. The state of the system, that is the values of the level variables, is determined at the times DT, 2 * DT, 3 * DT, ... This is continued until reaching the wanted time of consideration of the system, LENGTH.

Time is represented by TIME. TIME.K denotes the time, when the state is determined, TIME.J denotes the last time of determination, and TIME.L the next time of determination of the state:



This way of measuring the time is in DYNAMO represented by:

```
DYNAMO: SYSTEM DYNAMICS SYSTEM BEGIN
                                TIME: MEASURE BEGIN
                                          J, K, L: REAL
                                      END MEASURE;
                                DT, LENGTH: REAL;
                                .
                                .
                              END SYSTEM
```

The level variables are of the same type as TIME. The value of a level variable L at the time TIME.K is denoted by L.K and the value at time TIME.J by L.J:

```
TYPE LEVEL : MEASURE BEGIN
                  J, K : REAL
                END MEASURE
```

The rate variables define the changes of the level variables in time intervals of length DT, so they are described by the value in the interval from TIME.J to TIME.K (the JK value) and by the value in the interval from TIME.K to TIME.L (the KL value):

```
TYPE RATE: MEASURE BEGIN
                JK, KL : REAL
              END MEASURE
```

The differential equation

$$\frac{dL(t)}{dt} = R(t)$$

is in DYNAMO expressed by

```
L.K : = L.J + DT * R.JK
```

that is: the value of L at time TIME.K is the value at the last time of determination (L.J) plus DT times the change in the interval from TIME.J to TIME.K (R.JK).

Rate variables depending upon level variables at different points of time are expressed by e. g.

R.KL : = L1.K + L2.J

The state of the system at the times DT, 2 * DT, ... is computed by having the DYNAMO system repeat the following sequence of actions:

1. Increase TIME by DT; time of computation : = last time of computation + DT.

2. Compute the values of the level variables on basis of their values at the last time of computation and the changes in the interval from the last time of computation till now. Furthermore, compute the values of the rate variables of the next interval (the KL value).

   At the next time of computation, the level variables have "their values at the last time of computation" equal to the values at this point of time (TIME.K). So, before the time is increased by DT the following actions are to be executed:

3. for all variables:
   level variable.J is set to the K value,
   rate variable.JK is set to the KL value
   and further, TIME.J is set to TIME.K.

The repetition of this sequence of actions are terminated when the value of TIME.K has become greater than or equal to the value of LENGTH.

To perform the actions of 3. a DYNAMO system is described by the sets of level variables and rate variables.Quantities of the types LEVEL and RATE are automatically included in the proper set:

DYNAMO:
  SYSTEM DYNAMICS
    SYSTEM BEGIN
      SL denotes the set of level variables of the system,
      SR denotes the set of rate variables;

```
TYPE LEVEL: MEASURE BEGIN
                J, K, L : REAL;
                quantities of this type are elements
                of the set SL
              END MEASURE;

TYPE RATE: MEASURE BEGIN
              JK, KL : REAL;
              quantities of this type are
              elements of the set SR
            END MEASURE;
TIME : .....;
DT, LENGTH : REAL
L : QUANTITY OF LEVEL;
R : QUANTITY OF RATE;

OWN TASK BEGIN
    TIME.J : = 0;
    REPEAT
     (* TIME.K : = TIME.J + DT;
        compute the values of the level variables at the
        time TIME.K and the values of the rate variables
        in the interval KL;
        IF TIME.K ≥ LENGTH THEN TERMINATE;
        for all level variables L of SL:L.J: = L.K;
        for all rate variables R of SR:R.JK:=R.KL;
        TIME.J : = TIME.K
     *)
END TASK
END DYNAMO SYSTEM
```

The computation of the values of the level and rate variables is described by a sequence of DYNAMO-sentences, as e.g. "L.K : = L.J + DT * R.JK". This sequence characterizes the actual DYNAMO system and may therefore not be described in DYNAMO, which represents DYNAMO systems in general. Instead the imperative

"compute the values of the level variables ...."

is substituted by the imperative "INNER" :

```
REPEAT
  (* TIME.K : = TIME.J + DT;
     INNER
     IF TIME.K ≥ LENGTH THEN TERMINATE;
     :
  *)
```

In the description of a specific DYNAMO system (of the system class DYNAMO) "DYNAMO" is used as prefix. In the system object descriptor the specific variables and the sequence of DYNAMO actions are described, as e. g.

```
SDS: DYNAMO SYSTEM BEGIN
                   L1, L2 : QUANTITY OF LEVEL;
                   R1, R2 : QUANTITY OF RATE;
                   OWN TASK BEGIN
                      L1,K : = .....;
                      R2.KL : = .....;
                      :
                   END TASK
                 END SYSTEM
```

The attributes of the system object SDS consists of the attributes of DYNAMO and those described in the object descriptor. The own task of SDS is: the own task of DYNAMO with the sequence of DYNAMO actions described in the object descriptor substituting INNER . So the SDS is a DYNAMO system, where the values of level and rate variables are computed at the times DT, 2 * DT, ... according to the sequence of DYNAMO-sentences

```
L1.K : = .....;
R2.KL : = .....;
:
```

The values of DT and LENGTH and the initial values of the level variables are for the specific DYNAMO system set by:

```
SDS: DYNAMO SYSTEM BEGIN
                 :
                 END SYSTEM
                   PUT (* DT : = .....;
                          LENGTH : = .....;
                          L1.J : = .....;
                          L2.J : = .....;      *)
```

As mentioned earlier a DYNAMO system consists of only one component and thus has only one set of variables and one set of equations. When the mathematical model is to be represented by a DYNAMO system there must be introduced variables with different names for each of the components epilimnion and hypolimnion:

```
DYNAMO LAKE MODEL:
  DYNAMO SYSTEM BEGIN
           CPB, CNB, DPB, DNB, ACB, ZCB, FISKB, OOB):
           level variables of the hypolimnion:
             QUANTITY OF LEVEL;
           CPO, CNO, DPO, DNO, ACO, ZCO, FISKO, OOO).
           level variables of the epilimnion:
             QUANTITY OF LEVEL;
           EPTEM): temperature of the epilimnion,
           HYTEM): temperature of the hypolimnion:
             QUANTITY OF LEVEL
             :
       END SYSTEM
```

The rate variables are defined similarly.

The description of the same level variables and processes as treated in the section on the mathematical model will be a sequence of DYNAMO-sentences defining the relations between the variables:

```
DYNAMO LAKE MODEL:
 DYNAMO
   SYSTEM BEGIN
        :
        OWN TASK BEGIN
           COMMENT EPILIMNION;
           FACO.KL: = RPROD.K - ZACO.K - DACO.K;
           FZCO.KL: = VACO.K - DZCO.K;
           ACO.K: = ACO.J + DT * (FACO.JK + AACO.JK);
           ZCO.K: = ZCO.J + DT * (FZCO.JK + AZCO.JK);
           RPROD.K: = .....;
           ZACO.K: =  -KGO.K * ZCO.K;
           DACO.K: = KDAC * ACO.K;
           VACO.K: = VEFO.K * (KGO.K + 0.1 * DECONO.K);
           DZCO.K: = KDZCO.K * ZCO.K;
                     :
           COMMENT HYPOLIMNION;
           FACB.KL: =  -ZACB.K - DACB.K;
           FZCB.KL: = VACB.K - DZCB.K;
                     :
        END OWN TASK
   END SYSTEM
```

Chapter 2

On the Use of DELTA

The purposes of using the DELTA language for description of a lake model as in the preceding have been to investigate the use of the language as a tool of communication in an inter-disciplinary group and to investigate the use of the single DELTA concepts and language elements.

The following will therefore be in two parts:

1. Experiences with and an evaluation of the use of DELTA in the FSP-case, made in cooperation with the relevant FSP members.

2. My own experiences with the use of the single concepts and language elements.

Another purpose has been to investigate the need of a context concept. This is only briefly mentioned here. A more profound treatment will be the subject of a later report.

2.1 On the Use of the DELTA Language as a Tool of Communication

Part of the basis for the working out of the descriptions of the various models was to investigate the need of a common system description language in an inter-disciplinary project.

As a common "language" within FSP has been used mathematics (or mathematical notation). To work out the DYNAMO model the model group needed the results of the various groups in a mathematical form. Curves, which e.g. biologists use frequently, thus have to be expressed by a mathematical function. Besides mathematics there have been used schematic drawings, tables and diagrams showing the relations between the elements of the lake (e.g. Slutrapport 1973).

The experiences with the use of the DELTA language may be grouped as follows:

- A common frame of reference, as e.g. the 1.2 about the basic lake

- working out different descriptions for different purposes (and system generators),
  and/or
- working out descriptions of the concepts used.

The "compactness" of and the construction of DELTA descriptions make them stereotype to read, and predominance of passages with program-like descriptions have their impact on the "heaviness" of DELTA descriptions. But detailled DELTA description are bound to be "compact". The conclusion was therefore that descriptions in DELTA are only to be worked but in the cases where they are needed, that is by description of important system elements, where a precise description is necessary.

Further it was mentioned that it would help with a DELTA manual containing a.o. important concepts and key words and their definition. The observation that DELTA descriptions are to be read in an other way than normal prosa (e.g. the matching of parantheses, espicially OBJECT BEGIN and END OBJECT) lead to that a DELTA manual should be considered as an appendix (or supplement) to a pedagogical introduction to DELTA.

- On the question whether the DELTA language was considered as usable in description of models of this kind (biological, hydrological etc., which were not primarly the subject of study when developing the language) it was stated that it had not been as if the model was forced into certain lines because it should be described in the DELTA language.

- In developing DELTA descriptions it is essential with a good graphical illustration, this may in some cases be more or just as important as a good textual description. It is noted that only rules for illustration of the static structure of a DELTA system are developed, and it was suggested that rules for illustrating different forms of interaction between objects were developed. The alternative is that each DELTA system reporter develops his own rules.

- It is our experience that you ought to be aware of the fact that concepts, described in DELTA will be useful, both internally for co-ordination of the models of the groups and externally when presenting the model. The description might be more formalised than the one of 1.2, but not much more detailled.

- The need of a common language as DELTA will decrease with the degree of details in the work with the model. When treating details of a model there will be needed narrow, disciplinary concepts, which may be difficult to describe so formalized. It was stated that in general it may have a restrictive influence on the communication to be forced to learn a new language, and that it in all cases demands an extra effort to read DELTA descriptions, as they often will be more "semantic compact" and more strictly logically constructed than the descriptions, we usually read.

  It was intimated that a formalized language with "ridgid" elements migt have a restrictive influence upon especially voluntary, university-like research projekt, as e.g. FSP. In FSP the mathematical notation has in some case caused difficulties (e.g. differential equations), but the members of FSP are on the other hand not quite unacquainted with this notation. Problems of this kind may be solved by only referring to known (or described) concepts. A new system description language does not solve these problems in a better way.

  In connection with a discussion of external communication of the model (where the same qualifications as in FSP can not be assumed) mathematics and especially differential equations were regarded as well known. The problem with a alternative description as e.g. a DELTA description is that it can not be freed from e.g. differential equations, if it shall be as detailled as the mathematical model description. The solutions of these problems seem to be.

  - <u>descriptions at different levels</u>, where the first levels do not contain differential equations but e.g. changes in concentrations and the like, the gap between the introductionary descriptions and the detailled descriptions easily may become too big. But, this has nothing to do with the use of theDELTA language - also the descriptions in the FSP reports have this property.

As a conclusion you may say that all problems are not solved by "just using DELTA"; the way in which DELTA is used and the way DELTA is introduced are very important.

Besides the purpose of investigating the use of DELTA for communication the purpose of working out the DELTA descriptions has also been for me to become familiar with the FSP lake model, by describing in DELTA the models, I generated by reading the FSP reports.

The way of working implied by the use of DELTA has been appropiate when working with the model. The use of decomposition and the class/object concepts thus helped in the working out of the description of the basic lake concepts (1.2). The structure, which in this way was laid upon the talks and discussions about the basic components of a lake, helped us in understanding each other and in capteering and maintaining essential elements of the rather brain storming-like discussions.

The decomposition principle and the class/object concepts also helped me in reaching central aspects and problems of the model relatively quickly. The class concept has also in other connections been mentioned as a very usable concept in description of models of this kind. It shall be noted here that the DYNAMO simulation language has no class concept (not even a component concept). This may be the reason why the FSP lake model is a model with two water layers and not a general n-layer model.

DELTA in similar projects

The FSP project was at a rather far advanced stage when I started my work with DELTA descriptions of the lake model, so they have not been used directly in the project. But, if DELTA were used from the beginning of a similar project, how should it then be used?

We assume that the project will have a size and a progression as the one of FSP, starting with general meetings, where the problem is discussed and stated and where tasks for a number of groups are determined. The experience in FSP is that inter-disciplinary groups of that size can not communicate collectively and that the co-ordination is performed the best by a smaller group with representatives from the various groups.

A DELTA description of a common frame of reference - as e.g. that of 1.2 - would be usable. The description do not have to be more detailled.

The different groups shall in their work use this description and work out descriptions in DELTA of the most important parts of their work, especially the parts which are important in relation to the work of the other groups.

A joint model is assumed to be made in a special group, by composing the various DELTA descriptions. In cases, where the disciplines are related, this composition may be done without any big problems. In other cases it may be necessary to supplement the descriptions of the models with descriptions of the concepts used. These descriptions will also be usable in discussions of different understandings of parts of the model.

2.2 Experiences on the Use of the single DELTA Concepts

In this section I will treat the use of single DELTA concepts and my experiences on this.

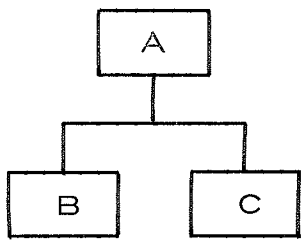
Graphical illustration

In the development of a DELTA system description and especially in the stages of decomposition there is a need of a way of illustrating nesting/splitting and a way of distinguishing between singular objects and classes of objects.

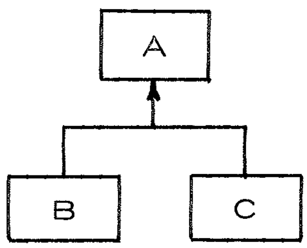
At the time when I needed this, there were only proposals of how to do it. Therefore I have used my own proposals.

Nesting

An object A containing the objects B and C (nesting) is illustrated by



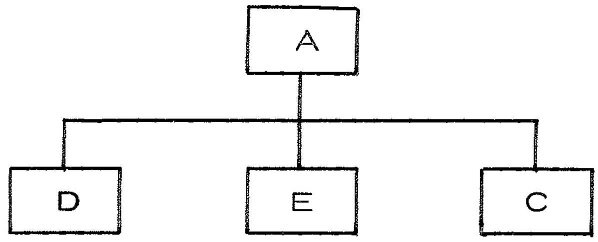
There is a drawback to this way of doing it and that is that the relative positions of the rectangles are important. This may be an advantage too, in that it limits the number of objects and the complexity of the structure of a figure; you are forced to divide the illustration in minor parts and thus obtain illustrations which are easier to capture. The alternative to this method is to illustrate the ENCLOSER reference, as e.g.



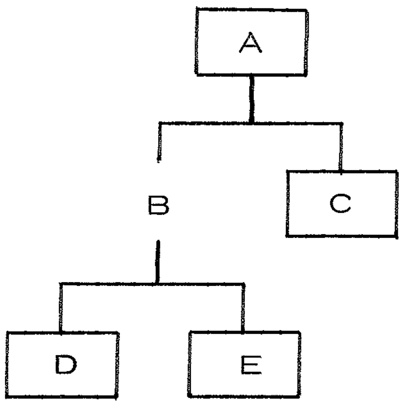
An advantage of this method is that illustrations may easily be extended with new objects.

Splitting

If B is to be split in D and E, then it might be illustrated by removing B from the figure:

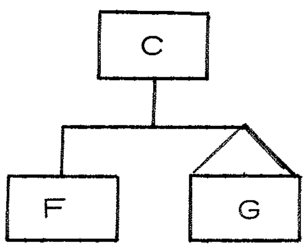


This would reflect that A contains no object B and that it is substituted by D and E. But, in the stage of decomposition, it will be an advantage to be able to recognize the structure of the preceding figures. So "B" should be part of the graphical illustration even if the object is splitted. My proposal is that the rectangle and not the title is removed when an object is splitted, as



Classes of objects

Until now only singular objects are illustrated. An object C containing a number of objects of a class G I have illustrated by:



This way of illustrating may be extended so that the number of objects is part of the illustration. In the triangle $\wedge$ "FIX" may denote that the number is fixed as long as the encloser exists and "VAR" that the number of objects varies. The number itself may be denoted by "5", "< 5", "> 5", or the like.

A disadvantage of this method is that class titles and object names are mixed with "FIX", "VAR" and numbers, and this may in extreme cases be confusing. This would also be the case if the indications of numbers were written inside the rectangles, together with the names and titles.

A solution of this would be to illustrate graphically whether the number is fixed or variable, as e.g.

<u>fixed</u> <u>variable</u>

or something like that. The indication of the number, that is "5" or "> 5" may be written inside the triangle and the half-circle.

<u>The development of a description</u>

To indicate that a description is introductionary and that more is to be added to the description I have done like this:

```
A:    OBJECT BEGIN
        X,Y : REAL;
        I : INTEGER;
        .
        .
        .
      END OBJECT
```

that is using dots to indicate possible description elements.

If more description elements are added to a description given before and if more is to be added, this is indicated by

```
A:   OBJECT BEGIN
          :
          FUNCTION F : REAL BEGIN ... END;
          :
     END OBJECT
```

And if no more is to be added to the description it is indicated by

```
A:   OBJECT BEGIN
          :
          OWN TASK BEGIN ... END TASK
     END OBJECT
```

These rules may be applied not only to object descriptors but also to e.g. imperatives, developed in a number of stages.

But these rules are only valid when applied to descriptions at the same level of abstraction. To obtain the possibility of indicating whether a description is an addition to or a refinement of an earlier description, the solution may be the introduction of system description level numbers. This is no new idea. In "Notes on Structured Programming" (Dijkstra72, pp. 26-39) Dijkstra suggests and gives an example of use of "Codes" to identify refinement of segments of program text. Knuth suggests in "Review of Structured Programming" (Knuth73, pp. 4-6) use of the class/subclass concept to indicate levels in the program (text). But this would in DELTA (and SIMULA) be a misuse of the class concept. As an example this method would not cover the refinement of the object descriptor of a singular object.

This may lead to a discussion of the relation between decomposition and system description. What is meant by working out descriptions in more levels?

There are two possibilities:

1. Make a description of the whole system every time a new layer of components has come into existance by decomposition.

2. Decompose the whole system as far as possible and then give a description in more levels of the components.

As far as it is possible it is preferable to follow the first possibility, as the structure of the description reflects (and is influenced by) the structure of the system. This has been tried in section 1.2 and the experience is that you only describe the new components. The second possibility has been used in description of the models of the various disciplines, with the implication that introductionary descriptions of the components are missing.

The two possibilities are only present when describing a model. When developing the model (also by describing it) the possibilities will not be used in their pure form. You will often "pursue" the decomposition of a component or a decomposition will take place after a detailed description of a component. The question is then: shall the result of a work with a model be a plain system description or shall it reflect the considerations made during the work?

The function concept

In the DELTA descriptions functions have been used to describe quantities, the values of which depend on the values of other quantities.

The value of a quantity X of an object A may in two ways depend upon other quantities:

1. the dependency is imposed by other objects and the definition of the value is thus part of the definition of these objects,

2. the dependency is part of the definition of A.

In the first case X will be introduced as a variable of A.

In the second case the dependency may be described by

- introduce X as a variable of A and define the value in an inner object or in the own task of A,

- describe X by a function.

In the cases where the dependency is a natural part of the definition of the quantity, that is where an inner object does not represent any component of the referent system or where the object is not naturally described by actions defining the value of X, the only alternative is to describe X by a function. It has thus been necessary to describe singular attributes (quantities) by patterns (for evaluations). Instead it should be possible to distinguish between quantities, the values of which depend upon other quantities, and patterns for evaluations.

The desired possibility is the possibility of describing the determination of the value of a quantity when describing the quantity. But, this is covered by the proposed measure concept, as the type of a quantity indicates a way of measuring some amount or quality, thus determining the value among the possible values of the type. An integer variable, I : INTEGER, will have its value determined by counting in some way. An integer variable, whose value depends upon other quantities, may be described by a singular measure:

```
F:    INTEGER MEASURE BEGIN
                    description of how to determine the
                    value of F among the possible values,
                    that is the integers
                END MEASURE
```

Functions will then be reserved for description of patterns for action sequences, resulting in a value and which normally will be executed in different contexts and with different values of the parameters.

The virtual concept and open descriptions

The virtual concept is in DELTA defined in connection with the subclass concept, making it possible to refer to elements, which in subclasses are defined differently. But it may also be used to indicate openings in a description. Virtual elements, which might have been defined in a subclass but which are not defined as the subclass is not defined, introduce an opening in the description. But shall this concept cover openings in descriptions of singular objects and in descriptions which are refined in a later stage of description?

In the description of HYDROLOGY/HYDRAULICS LAKE I have not introduced the functions STREAMING and PERMEATION INWARDS/OUTWARDS of the singular objects SURFACE, SATURATED LAYER and UNSATURATED ZONE as virtuals. This is done because I think that VIRTUAL ought to be reserved for elements, which may be defined in subclasses, and that a new concept covering openings in a description should be introduced. And this new concept should not only cover openings in descriptions of singular objects, but openings in description elements in general.

I have not proposed any keyword to indicate that the specification of an element is open, as it is a question whether it is neceasary or not. Open descriptions have been used since the beginning of the development of DELTA. A description as

```
SYSTEM BEGIN
        the system consists of
        A : OBJECT ;
        B : OBJECT ;
        C : OBJECT ;
        and is furthermore characterized by ...
END SYSTEM
```

contains open descriptions of A, B and C. Similarly I have specified "STREAMING : FUNCTION" , and in other cases you may specify:

```
X,Y : QUANTITY ;
TP : TASK PROCEDURE;
```

The system class concept

In the DELTA descriptions I have implied the existence of a system class concept. The use of this concept will have its limitations if it is only quantities and references of the system object head which may be initialized by generation.

In section 1.3 the TOPOGRAPHY/MORPHOGRAPHY LAKE is introduced as a system class, and it is shown how quantities of the system object (POSITION and ALTITUDE) are initialized. But there is also a need for initialization of quantities of the singular objects SURFACE and WATER MASS of the system object.

A solution would be to provide the system object head with "initialization quantities", corresponding to the quantities of the singular objects. These may then by generation of the singular objects be initialized to the values of the quantities of the system object head. In this way there would be nothing gained by decomposing a system (except, of course, to be able to describe concurrent actions).

As the singular objects of a system are generated at the same time as the system object, there should be no problems in initializing these when initializing the system object. For objects, which are generated after the generation of the system object, this cannot be done without changing the structural properties of DELTA systems,as e.g. the property that the generator of an object has not qualified access to internal objects of the generated object. For the system object, which is not generated by any object but by the system generator, this property hoes not have to be fulfilled.

Generalized type concept

Early in the work with the DELTA descriptions there was a need of other types than those given in DELTA . So I have implied the existance of a generalized type concept and I have - without going in details - described types and declared quantities of these types. My intention in using a generalized type concept has been to make the descriptions easier to understand. The types of DELTA would have been sufficient, but then the descriptions would not have been so good.

The first use of something besides the types of DELTA were "open" descriptions as e.g.

```
DEPTH : QUANTITY
```

and descriptions of data structures as

```
DEPTH : QUANTITY BEGIN
            UPPER LIMIT, LOWER LIMIT : INTEGER
        END QUANTITY
```

This way of consideration focused on the single quantity. The starting point for development of a generalized type concept ought to be that a type is defined by a set of possible values and a set of operations upon these. The use of a generalized type concept has been mainly to describe different ways of measuring something, e.g. biomass, concentration, rate of flow, biomass change etc.... This and more fundamental considerations lead to the measure concept as described in DELTA 74. This way of considering types and quantities has been natural in this context. Even though details of a type concept have not been used it has been clarifying to be able to distinguish between quantities of different kinds by defining different types - as an example take the types biomass and biomass change. In order to obtain the same without a type concept every quantity declaration should be supplemented by a comment.

Chapter 3

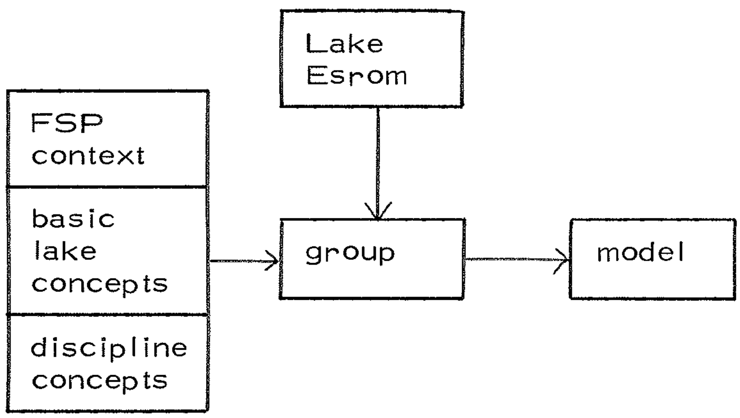
The Aspects of a Context Concept

In this report DELTA is used for system description in a situation where the same system is viewed from different points and where the goal has been to get a joint model of the system. In this connection the need for a context concept has been illustrated.

As a reflection of how it has been done in FSP, a DELTA description has been worked out for each group.

Within the single group the existing DELTA concepts are sufficient: one part of the world is considered as a system and a model of it is made. In order to work in a project with other groups there has been need of including elements, which - viewed from the single group - are regarded as a part of the "environment" of the group:

- Working out of the model: This has been done in a context defining the degree of detailing and the basic components of a lake model, common to all groups. To illustrate this the context concept as in DELTA has been used. Every group has as context the collection of the "FSP context" (that is the goal and purposes of FSP), the basic lake concepts and components (the frame of reference, as described in 1.2) and the concepts of the discipline(s) represented by the group. The illustration of this has been:



- Communication of the model: For internal communication there has been a need for - besides the system description - a description to the other groups of the concepts used. This will be the case too for the model group (and the project as a whole) when externally communicating the joint model.

I have not tried to define the context concept further than in DELTA74, but I have described and used contexts. The use has been as with system classes of SIMULA: In a system description the title of a context may be used as a prefix, as e.g.

BIOLOGICAL LAKE:
BIOLOGY SYSTEM BEGIN ..... END SYSTEM

On the other hand, there has also been need of a system class concept, for description of a general lake model. In the treatment of the mathematical model and the DYNAMO model I have also used a system class concept and suggested the difference between the two concepts. Whether it is reasonable or not to introduce a system class concept must be decided in connection with the discussion of the introduction of a context concept.

The purpose of a description of a context should be to supplement a system description. In this way it should be possible to give the system generator a better understanding of why the model and its description is as it is. A description of a context in this situation might have the form:

```
HYDROLOGY/HYDRAULICS:
 CONTEXT BEGIN
        A hydrological/hydraulic model of a lake (or aquatic
        system) has  the following aspects: .....

        and has as purpose: .....

        The basis of a lake model is the components and
        concepts described in "Basic lake concepts".

        In describing a lake the following concepts are used:
             TYPE RATE OF FLOW: .....;
             TYPE RAIN GAUGING: .....;
             CLASS WATER LAYER: .....;
             :
```

Furthermore the following concepts from
TYPOGRAPHY/MORPHOGRAPHY are used:
<u>TYPE</u>s LINEAR MEASURE, AREA MEASURE
and VOLUME MEASURE

<u>END</u> <u>CONTEXT</u>

Descriptions like this are supposed to exist for TYPOGRAPHY/ MORPHOGRAPHY and for BIOLOGY, the last with a class WATER LAYER with another definition than that of HYDROLOGY/HYDRAULICS.

The joint lake model will be made in a context LIMNOLOGY, which is a composition of the relevant contexts and which may be used for description of other lakes or aquatic systems. By this composition of context there may be need of modifications of the resulting context, as e.g. with the class WATER LAYER.

The preliminary use of a context concept in this report has shown some of the aspects of and desired properties of a context concept. Yet many problems are not discussed. On one side the concept is needed to cover the single system generator's interpretation of information. On the other side it has also to do with elements which may be used by more than one system generator and in descriptions of several systems. Is a context associated with the single system generator or with a category of system generators? Which parts of the "environment" is it to cover? Is a context an explanatory text following the system description or is the concept of a context description to be introduced? Do concepts (as e.g. those of System Dynamics)exist without the description of them? And is a context to represent something physically existing or "just" concepts?

The answering of all these questions - a definition of a context concept in DELTA - may be done after an analysis of the concept in the general system description situation and in other connections. This will be the subject of a later report.

References

DELTA 74:

Kyng, M & Pedersen, B.M.:
"Description of a Model of a Single Helix Pomatia Brain Neuron and an associated Neurophysiological Experiment".
DELTA Project Report No. 3, DAIMI PB-42, December 1974

DELTA 75:

Holbæk-Hansen, Håndlykken, Nygaard:
"System Description and the DELTA Language" -
Delta Project Report No. 4, Norwegian Computing Center, September 1975

DIJKSTRA 72:

Dijkstra, E.W.:
"Notes on Structured Programming"
In Structured Programming, Academic Press 1972

FSP 75:

Fælles Sø Projekt - Slutrapport 1974
Maj 1975

GORDON 69:

Gordon, G:
"System Simulation"
Prentice-Hall, Inc. 1969

HEGNA 74:

Hegna, H.:
"DYNAMO in SIMULA 67 - A rough Outline of a Simple Implementation"
Norwegian Computing Center 1974

KNUTH 73:

Knuth, D. E.:
"A Review of "Structured Programming" "
STAN - CS - 73 - 371, June 1973

PUGH III 63:

Pugh III, A. L.:
"DYNAMO - Users Manual"
The M. I. T. Press 1961 & 1963.