

Contents of DESERT HELP

A general overview of the organization and features of DESERT 1.0

The representation of riverine systems and their characteristics, the input data, the methods for performing hydraulic and water quality simulations, calibration of water quality models, and optimal wastewater treatment plant allocation, and the export of data

[A general description of DESERT 1.0](#)

Working with DESERT 1.0

Building the input data files, defining the river network of interest, choosing the methods for carrying out hydraulic and water quality simulation, defining the water quality models, exporting simulation results, calibrating the water quality models, and optimizing wastewater treatment allocation

[Working with DESERT 1.0](#)

Menus, commands, and tools of the DESERT window

The menus, commands, and tools available in the DESERT window for dealing with files, defining the riverine networks of interest, etc.

[DESERT window](#)

The simulation language MODUS

The elements and syntax of MODUS, the language for defining water quality models, calibration and optimization procedures, and operational instructions

[MODUS language](#)

Error messages

The error messages produced by DESERT

[Error messages](#)

A general overview of the organization and features of DESERT 1.0

The representation of riverine systems and their characteristics, the input data, the methods for performing hydraulic and water quality simulations, calibration of water quality models, and optimal wastewater treatment plant allocation, and the export of data

The representation of the riverine systems and of their characteristics (effluents, measurement points, confluences, weir, etc.) in DESERT 1.0

The riverine systems and their characteristics

A general overview of DESERT 1.0. Its organization and its performances.

Overview of DESERT 1.0

The riverine systems and their characteristics

The representation of the riverine systems and of their characteristics (effluents, measurement points, confluences, weir, etc.) in DESERT 1.0

The riverine systems as bynary trees

Structural objects

River objects

The riverine systems as binary trees

In DESERT, a riverine system is organized as a binary tree made of reaches. Reaches are segments of the rivers forming the system, where hydraulic and water quality properties are in some sense uniform. Points of separation divide reaches from each other. The fact that the system is organized as a binary tree implicates that, although branching is allowed, no meandering or circling is possible. Also, by convention, the root of the binary tree must be the downstream end of the river system. As a consequence, no bifurcation of rivers is allowed. Another restriction on the topology of the system is that, at each confluence point, only two reaches can merge to form a new one.



Figure 1 Admissibility of riverine system schemes.

a) admissible, since it is a binary tree; b) not admissible because showing a confluence with three merging branches; c) not admissible because having two roots and a bifurcation; d) not admissible, because the presence of a cycle within the tree. The arrows indicate the main direction of flow.

The fact that the riverine system is treated as a binary tree implicates that an order is associated to each reach of the system. Adjacent reaches of the same order form a branch of the river system. When two reaches merge at a confluence, one keeps the same order of the downstream reach (therefore it belongs to the same branch of the downstream reach). The other reach increases its order of 1, thus beginning another branch (**Figure 2**).

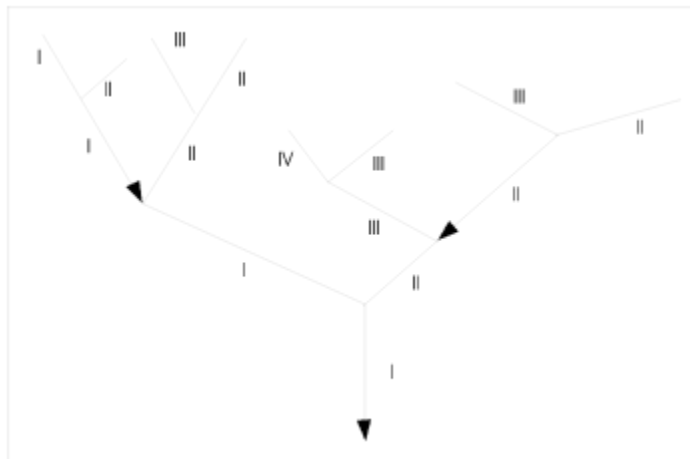


Figure 2 Representation of a riverine system as a binary tree.

Arrows indicate the main direction of the flow. Numbers indicate the order of reaches.

Characteristics of riverine systems (hydraulic sections, confluence of rivers, effluent point sources, lateral inflows, measurement points, etc.) are expressed as a collection of objects belonging to certain classes and situated along the reaches. There are two kinds of

objects: **structural** objects and **river** objects.

Structural objects are objects that modify the topology of the hydraulic system. They define the beginning and the end of reaches. They do not have any information record associated to them.

River objects are objects that do not modify the topology of the system, but have one or more information record associated to them.

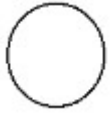
Each class is identified by a unique operational code.

See also

[Structural objects](#)

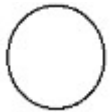
[River objects](#)

Structural objects



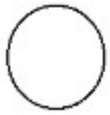
Point of separation of reaches. It indicates the end of a reach and the beginning of a new reach. Reaches are segments of the river uniform from the hydraulic and water quality viewpoint. Typically, a reach can be a segment of a river where water discharge is longitudinally uniform, a segment of a river upstream or downstream a load emission, and so on. DESERT automatically generates a Reach cross section object in correspondence of this object. Its shape is obtained by linear interpolation of the two nearest Reach cross section objects present on the same branch. This is done to supply boundary conditions for hydraulic simulation.

The operational code identifying this class is R



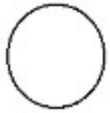
Confluence of two reaches. This object indicates that two reaches merge to form one reach only. Water discharges of the merging reaches are summed to produce the water discharge of the reach downstream of the confluence. A Confluence object automatically acts also as a Point of separation object. DESERT automatically generates a Reach cross section object in correspondence of this object. Its shape is obtained by linear interpolation of the two nearest Reach cross section objects present on the same branch. This is done to supply boundary conditions for hydraulic simulation.

The operational code identifying this class is C for the secondary reaches intersecting the main branch on the right and CL for the secondary reaches intersecting the main branch on the left.



Headwater point. A Headwater object must mark the upstream end of each terminal reach. DESERT automatically generates a Reach cross section object in correspondence of this object. Its shape is obtained by linear interpolation of the two nearest Reach cross section objects present on the same branch. This is done to supply boundary conditions for hydraulic simulation.

The operational code identifying this class is H



End point. An End object must mark the root of the binary tree representing the system. DESERT automatically generates a Reach cross section object in correspondence of this object. Its shape is obtained by linear interpolation of the two nearest Reach cross section objects present on the same branch. This is done to supply boundary conditions for hydraulic simulation.

The operational code identifying this class is E

RIVER OBJECTS



Lateral inflow source. The presence of this object means that there is lateral inflow along the reach where the object is located. The inflow rate is specified in $m^3/s/Km$. One or more water quality records are associated to this object (**see Chapter 3.3**).

The operational code identifying this class is I



Water intake. This object indicates a water intake from the reach. It can be associated to one or more discharge and water quality records (**see Chapter 3.3**). Discharges are expressed in m^3/s and water quality data in mg/l .

The operational code identifying this class is A



Measurement point. It represents a point where water quality has been measured. It can be used both for representing available data and for calibration purposes. One or more water quality records are associated to this object (**see Chapter 3.3**). Water quality data in mg/l .

The operational code identifying this class is M



Reach cross section. This object indicates the shape of the reach cross section in a point. A succession of pairs distance from the longitudinal axis of the reach-elevation of the riverbed above mean sea level describe the cross sectional shape. Since the hydraulic and water quality models are one-dimensional, the shape of the cross-sections is supposed symmetric with respect to the longitudinal axis. For this reason, it is necessary to specify the pairs width-bottom elevation of half the section only. DESERT automatically associates a Reach cross section object to every structural object. Its shape is determined by linearly interpolating the two nearest Reach cross section object of the same branch. Both width and bottom elevation are expressed in meters.

The operational code identifying this class is F



Water quality constraint. This object indicates that water quality should satisfy some constraints at the location of the object. It has associated one or more water quality records expressed in mg/l (**see Chapter 3.3**). It is used mainly for optimization purposes.

The operational code identifying this class is Q.



Point Source. This object indicates that some effluent is discharged where the object is located. One or more water quality records describe the quality and quantity of the effluent (**see Chapter 3.3**). Discharges are expressed in m^3/s and water quality data in mg/l .

The operational code identifying this class is S.



Weir. This object indicates the presence of a hydraulic constraint to the flow. The constraint can be either a discharge-elevation relationship or a water surface elevation-time relationship. (**see Chapter 3.3**).

The operational code identifying this class is W.



Treatment plant. This object identifies a treatment plant discharge. Water quality resulting from alternative policies and the relative costs can be associated to this object (**see Chapters 3.2 and 3.3**).

The operational code identifying this class is T.

Their logical position must be given in the River Index Code, a system based on the concept of binary tree and of order of the branches (**see Chapter 3.3**).

WARNING

WARNING

Any of the River object can be associated to a structural object if they need to share the same location. For example a location can be at the same time a Source and a Reach point; in this case the code will be SR. Other frequently used combinations are:

- Measurement and Headwater point (code MH);
- Measurement and End point (code ME);
- Source and Reach point (code SR);
- Confluence and measurement point (code MC)

Of course two structural objects can not be associated together (e.g., the code RC can not be accepted). If two or more river object need to share the same position, than they need to be associated to a structural object. For example, if an abstraction point and a measurement point are located at the same position along a reach, then the first must be declared as AR and the second as MR.

<p>The Treatment plant, the Water quality constrain, and the Weir object must be associated to a structural object. Their presence changes drastically the way hydraulic simulation or optimization are performed. If the codes W, T, or Q are used alone, the relative objects will not be considered. Therefore, for example, the code for a treatment plant must be TH if it is associated to a headwater, TC if it is associated to a confluence, TR if it is associated to a point of separation of reaches, or TE if it is associated to the root of the system.</p>

A general overview of DESERT 1.0

DESERT 1.0 has been written completely in C++ language to permit the use of the object oriented programming and the possibility of management and upgrade of the code that this language allows.

Figure 3 depicts the overall scheme of the software.

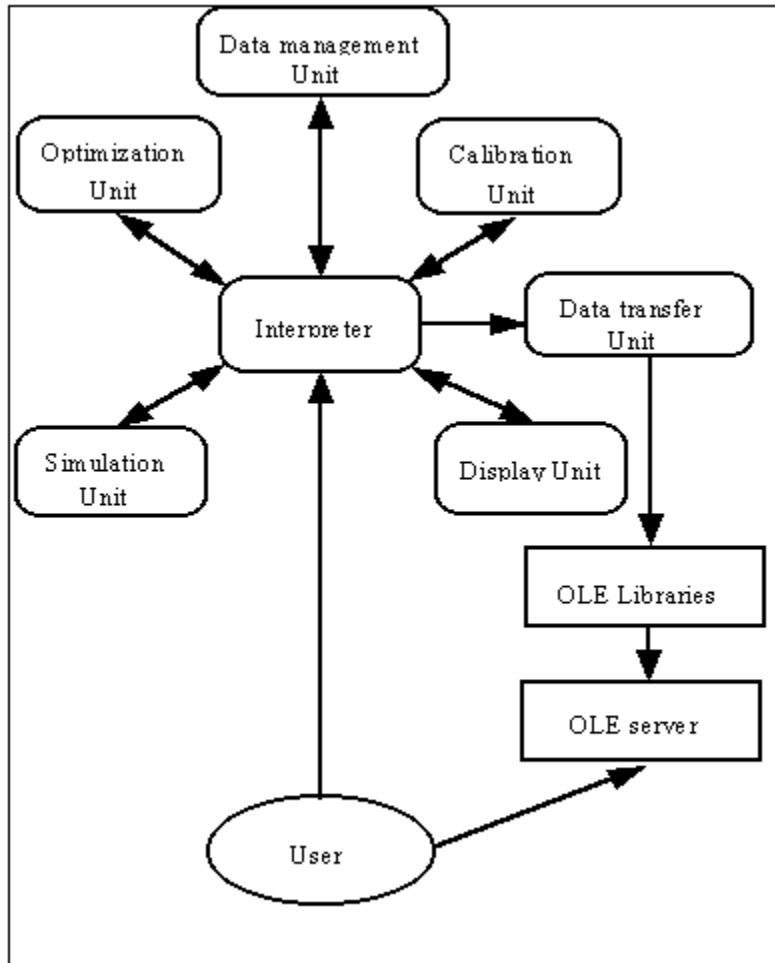


Figure 3 Outline of the DESERT 1.0 software.

A deeper description of the single sections is given later in [Data management unit](#)

[Display unit](#)

[Hydraulic unit](#)

[Water quality simulation unit](#)

Data transfer unit

Calibration unit

Optimization unit

Data management unit

One of the main features of DESERT is that it incorporates a powerful dBase style relational database engine to handle input data. This feature allows the user to treat the large amount of heterogeneous data (data on the river network, effluents, treatment plants, monitoring network, etc.) necessary for a river basin scale assessment using any dBase compatible database management software. In this way the user can organize input data in a really effective, flexible, and easy way (**See Chapter 3.3**). The fact that monitoring networks often organize their results using databases is a further element in favor of this choice. Also it is very easy the convention from the output format of traditional programming language (normally text separated by commas or tabs) into dBase. The only limitation of the dBase engine is that it is based on the dBase III language. The use of later versions of this language (as the more common dBase III+ or dBase IV) is possible most of the times. However, since some problems could occur, it should be better to use the version III of dBase.

Display unit

As already introduced, river basins consist of spatially distributed objects, such as river stretches, gauging and effluent source points, sampling locations, weirs, etc. Therefore, the best way to represent such a collection of objects is to display a scheme of its spatial distribution. The display unit guarantees this possibility. The display unit draws river basins in a symbolic way and is capable of scaling, scrolling and selecting particular river objects.

DESERT can automatically generate the representation of the river basin as a binary tree in a very schematic way. Reaches belonging to the same branch are represented along horizontal or vertical lines. At the confluences, the lower order branches depart from the main order branch perpendicularly. The user can only specify if the lower order branch intersects the right or the left side of the main branch. Vertical representations of vertical branches are proportional to each other. On the contrary, drawing requirements determine the length of every single horizontal branch.

It is also possible to have a representation of the system closer to reality. This can be done by supplying the geographical coordinates of the structural objects as vector files in the format of commercial MapViewer (TM) mapping software from Golden Software, Inc.. This format can be easily produced manually or adapted from other formats if needed (**See Chapter 3.3**). In this case, the length of each reach is proportional to the difference between the geographical coordinates of its boundary points and not to the actual distance.

Hydraulic unit

Computing river hydraulic parameters, such as depth, cross-section area, travel time, etc., is a necessary precondition for the simulation of water quality. The hydraulic models for rivers and open channels are based on mass continuity and moment equations from fluid mechanics (Antontsev et al., 1986). Most of the hydraulic and water quality simulation packages oblige the user to adopt the same method of solution for all the elements of the network. This restriction makes the software simpler, but does not take into consideration that available data and users interest vary from reach to reach. On the contrary, DESERT allows the use of a specific hydraulic model for every reach of the river basin. A dialogue box supplied by the display unit (**see Chapter 3.5**) allows the user to choose the hydraulic model out of the five possible alternatives listed below. However, all the five alternatives are based on a one dimensional spatial approach. That is, water velocity is averaged over the cross-section area of the reaches.

None

Water balance (No mesh)

Water balance (Mesh)

Steady state approximation

Diffusion wave approximation

WARNING

WARNING

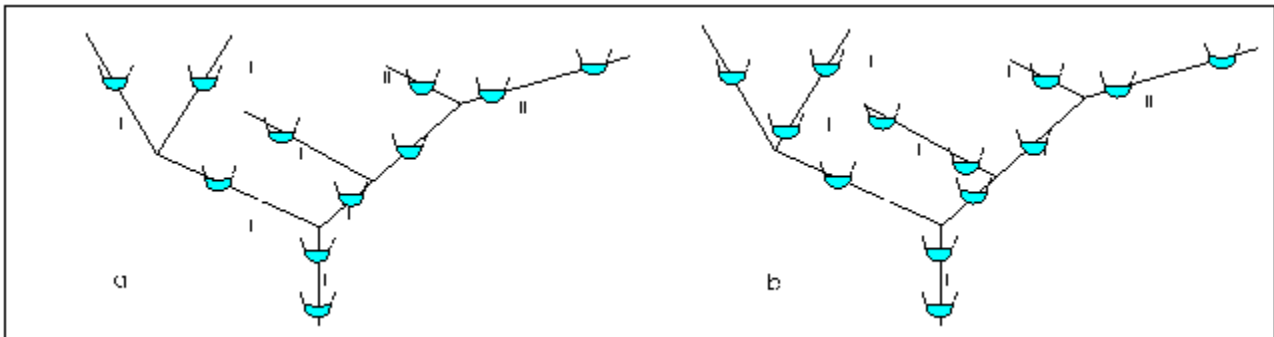


Figure 4 Requirements for the application of Steady state and Diffusion wave approximations.

a) Steady state or diffusion wave approximation can not be applied to the whole system (the branch of order II on the left and one of the branches of order III on the right have only one section); b) Steady state and diffusion wave approximation can be applied to the whole system (all the branches have at least two cross-section)

A minimum number of two Reach cross section objects is necessary for any branch where steady state or diffusion wave approximations are used. If this requirement is not satisfied, DESERT will not have sufficient information to carry out the hydraulic simulation (**see Figure 4**). On the branches where this minimum requirement is not satisfied Water balance methods can be applied. The above mentioned is a minimum requirement for using the momentum equation. However, to have a reliable hydraulic simulation, experience points out that a cross section should not be distant from the next cross section more than five times the space equivalent to its width.

None

No hydraulic simulation is performed. Therefore, the reach is considered as not existing.

Water balance (No mesh)

The objects present on the reach subdivide the reach itself into segments. Simple steady state water balance is performed between consecutive segments. Consequently, the hydraulics is described by the continuity equation alone in the following form:

$$Q_d = Q_u + q$$

where **q** is positive if the object discharges water in the reach or negative if it withdraws water from the reach

Q_d is the discharge downstream the object

Q_u is the discharge upstream the object

This method, and the **Water balance (Mesh)**, are typically used for reaches where unimportant processes take place, for reaches where hydraulic data are not sufficient, or during early stages in the modeling process.

Water balance (Mesh)

This alternative uses very similar to the **Water balance (No mesh) method**. The difference is that the reach is subdivided into segments of length specified by the user and water balance is not applied from object to object, but from mesh point to mesh point.

$$Q_{n+1} = Q_n + q$$

where **q** is positive if water is discharged between points n and n+1 or negative if it is withdrawn

Q_{n+1} is the discharge at the mesh point n+1

Q_n is the discharge at the mesh point n

Steady state approximation

For a steady-state situation, which is typical of low-flow periods, the momentum and continuity equations can be simplified by omitting terms that are responsible for the dynamic behavior of the flow. A quadratic law of resistance usually represents the friction term in the momentum equation:

$$\frac{\partial u}{\partial x} = - \frac{\alpha |u|}{K^2}$$

while continuity equation is given by:

$$\frac{\partial Q}{\partial x} = q(x)$$

where x is the co-ordinate along the river or channel

Q is the stream flow rate

h is the local depth of the river or channel

K is the resistance parameter (that in DESERT has a Mannings form)

q is the lateral inflow

Of course, also in this case the reach is subdivided in a series of segments, the length of which is specified by the user through a dialogue box (**see Chapter 3.5**). The solution of this system of ordinary differential equations is carried out with a Modified Euler Method.

Diffusion wave approximation

Since the software is oriented primarily towards water quality management, the level of complexity of hydrological models can be limited. For dynamic situations (non-stationary in time), the diffusion wave approximation (Antontsev et al., 1986) strikes a good balance between complexity, accuracy, and computation speed:

$$\frac{\partial z}{\partial t} + \frac{\partial Q}{\partial x} = q(x,t)$$

$$\frac{\partial z}{\partial x} = -\frac{Q|Q|}{K^2}$$

where z is the water surface elevation

q is the water discharge source along the river stretch

A is the cross-sectional area along the reach

t is the time variable

The method of solution of this couple of partial differential equation is a fully implicit scheme with Newtonian iterations. Steady state simulation with the initial boundary conditions determines the initial condition for the dynamic method. The boundary condition normally adopted for the downstream nodes is

$$\frac{\partial z}{\partial x} = 0$$

However, if the downstream node is a Weir, then a $Q(z)$ relationship or a $z(t)$ relationship can also be imposed.

Water quality simulation unit

Water quality assessment in riverine systems is generally carried out solving a set of ordinary or partial differential equations, which describe relevant physical and chemical processes. Most of the water quality software packages implement only one method of solution of the transport equation for all the network. Often, they are also problem specific (i.e., they deal with eutrophication or with heavy metals, or with pesticide, etc.). Also they often oblige the user to adopt one formulation of specific processes (light extinction, temperature dependence, photosynthetic rates, and so on) among the ones supported by the software library. Alternatively, the modeler must code the model in some computer language, as FORTRAN, C, or PASCAL, build the executable module with the help of tools like compilers, linkers, librarians, etc., and run it. The process is iterative, since, after evaluation of results, corrections may be needed to programming script or model formulation. The overall process is very slow and prone to errors (see **Figure 5**).

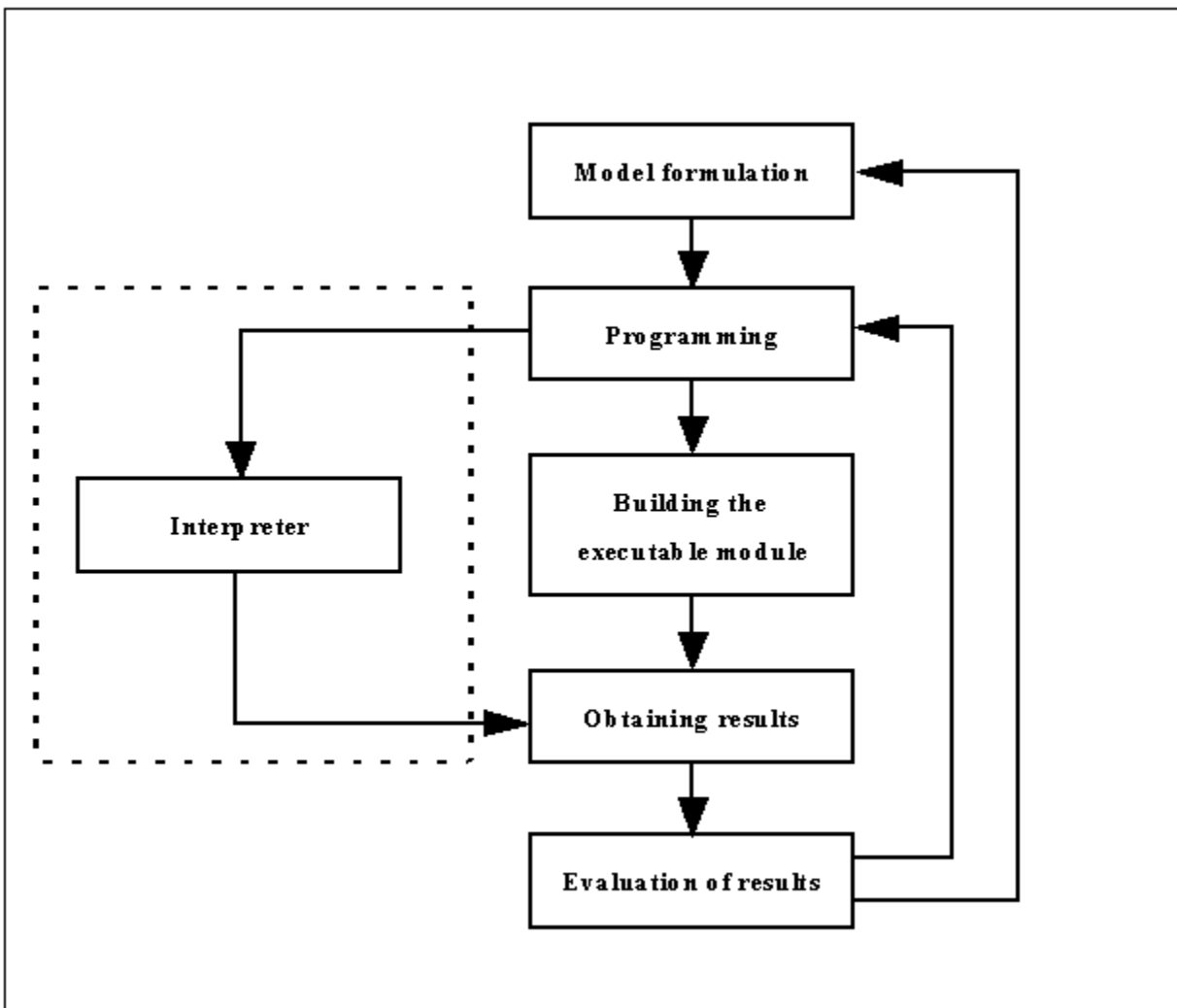


Figure 5 Conventional modeling approach (right) versus DESERT approach (left, in the dashed box).

DESERT represents an alternative to both the previously illustrated approaches. First, it supplies five different ways to simulate mass balance equations that may coexist in the same river basin schematization. Second, and more important, for three of these approaches (Matter kinetics (No mesh), Matter transport (Steady state solution), and Matter transport (Dynamic solution)) description of reaction processes can be given through an interpreting language similar to BASIC. However, to speed up the simulation process, the transport part of the equations is precompiled. In fact, a mass conservation equation for riverine systems can be expressed as follows:

$$\frac{\partial C_i(x,t)}{\partial t} + \frac{\partial C_i(x,t)}{\partial x} = q(x,t) \cdot C_i(x,t) + F_i(x,t, C_1, C_2, \dots, C_N)$$

$i=1, \dots, N$

where: C_i are the water quality constituents (state variables)

N is the number of state variables

q_i is the concentration of i th water quality constituent in source water discharges

F_i is the function that expresses the reactions scheme of the i th state variable

Water quality equations can be decomposed into two sections:

1. the left hand side, which is the generic part of the equation (transport equation) and that is computed using precompiled instructions;
2. the right hand side, which is the specific part of the equation (reaction scheme) that is computed using interpreted commands.

This approach, together with the way input data are treated, allows the user to be free in describing whatever dynamics he (or she) is interested on. That is, the user can specify as many variables as desired and define whatever reaction may be needed. However, the user should note that among the arguments of the function

$$F_i(x,t, C_1, C_2, \dots, C_N)$$

derivatives do not appear. This is the only limitation to the formulation of the right hand side of the transport equations. The most important consequence of this limitation is that longitudinal dispersion is not considered. On the other hand, in most of the riverine systems, advection is the dominating factor of transport phenomena (Sómyódy and Varris, 1992).

In specifying the reaction schemes of the model and in determining the state variables, the user must consider that the only one water quality model is applied to all the reaches composing the system. Only the method for solving the mass balance equation can vary from reach to reach.

Another aspect to be considered is that all the six alternative water quality formulations are based on a one dimensional spatial approach. That is, water quality variables are averaged over the cross-sectional area of the reaches. The method for solving the water quality equations implemented in DESERT are the following

None

Matter balance (No mesh)

Matter balance (Mesh)

Matter kinetics (No mesh)

Matter transport (Steady state approximation)

Matter transport (Dynamic approximation)

None

No water quality simulation is performed and the reach is considered as not existing from the water quality view point.

Matter balance (No mesh)

Simple steady state mass balance is performed on the segments of the reach delimited by the objects present on the reach itself. The element concentrations are described by continuity equation alone in the following form:

$$Q_d C_d = Q_u C_u + qC$$

where Q_d is the discharge downstream the object

Q_u is the discharge upstream the object

q is positive if the object discharges water in the reach or negative if it withdraws water from the reach.

C_d is the concentration downstream

C_u is the concentration upstream

C is the inflow concentration.

This method, and the **Matter balance (Mesh)** method, are typically used for the reaches where unimportant processes take place, for reaches where water quality data are not satisfying, or during early stages in the modeling process.

Matter balance (Mesh)

This alternative is similar to the **Matter balance (No mesh)** method. The difference is that the reach is subdivided into segments of length specified by the user (the same used for the solving hydraulics, if a meshed method was chosen). Then, water balance is not applied from object to object, but from mesh point to mesh point.

$$Q_{n+1}C_{n+1} = Q_nC_n + qC$$

where Q_{n+1} is the discharge at the mesh point $n+1$

Q_n is the discharge of the mesh point n

q is positive if water is discharged between points n and $n+1$ or negative if it is withdrawn

C_{n+1} is the concentration at the point $n+1$

C_n is the concentration at the point n

C is the inflow concentration

Matter kinetics (No mesh)

With this option, no differential or transport equation is solved. The user directly specifies the desired kinetics of variables. Exponential, Bessel type, ERF type, and other kinds of behavior can be used for describing element kinetics. Functions are applied in the segments between objects. The contribution of inflow or withdrawn to the mass balance is calculated at the object positions. The selected behavior is then applied to the new concentration, i.e.:

Along the segments

$$C_i(x) = F_i(x, C_1, C_2, \dots, C_{i-1}, \dots, C_N)$$

At an object of position x_0 :

$$Q(x_0+)C(x_0+) = Q(x_0-)C(x_0-) + q + C+$$

where x_0+ is a position immediately downstream the object
 x_0- is a position immediately upstream the object.

Matter transport (Steady state approximation)

The steady state approximation of transport equations solves the following equation:

$$\frac{\partial(Q \cdot C^i)}{\partial x} = q^+(x) \cdot C_i^-(x) + F_i(x, C_1, C_2, \dots, C_i, \dots, C_N)$$

The reach is subdivided in a mesh of points. The mesh size is introduced by the user (**see Chapter 3.5**). The numerical method used to solve the equation is a Runge-Kutta order IV.

Matter transport (Dynamic approximation)

A set of full transport equations is solved in this method for the points of a mesh of user defined dimension (**see Chapter 3.5**). The initial values are determined either by interpolating input data files' values or by running the Matter transport (Steady state solution) methods with the initial boundary conditions. The solution of the system of equations is obtained by an implicit 1st order running wave method.

Data transfer unit

Simulation results have to be presented in some suitable way: as table, chart, or plot. It is not a simple task, since simulation data are usually multidimensional. Moreover, results often must undergo some postcomputational processing, such as statistical analysis or curve fitting. Commercial spreadsheet packages such as Microsoft Excel, Lotus 1-2-3, Corel Chart, etc., can handle this task very easily. The traditional approach of water quality and hydraulic simulation packages has been to produce results in a text file form, with the fields separated from each other by commas, tabs, or spaces. These text files were later processed by some spreadsheet or statistical package, a procedure slow and prone of errors of conversion. Thanks to the Microsoft Windows 3.1 data transfer mechanism based on OLE (Object Linking and Embedding) protocol, today it is possible to establish linkages between applications. Thus, DESERT software makes use of OLE server applications and can save and plot results directly to a spreadsheet. Although theoretically any spreadsheet implementing OLE protocol could be used, in practice only Microsoft Excel is able to satisfactorily handle data produced by DESERT. Data transfer unit facilitates transfer of simulation data through OLE libraries to Microsoft Excel, where data can be independently processed, stored, plotted, and so on.

Calibration unit

Calibration procedure is especially important in water quality management, since in the state-of-the-art approach all the uncertainty associated with the modeling process is treated as parameter uncertainty. A number of methods have been developed for parameter estimation and uncertainty analysis in water quality modeling. Most of them use minimization procedures with corresponding loss function such as the least squares methods (Beck, 1979a). Recursive methods, such as Kalman filter, have been also applied (Beck, 1979b; Rinaldi et al., 1979).

All these traditional methods produce a single best set of parameter values that is then used for the simulation. Recent developments pointed out that the estimation of a single set of parameters is not able to account for all the uncertainty peculiar to water quality modeling. The stochastic methodology for parameter estimation currently implemented in the software is the Hornberger-Spear-Young behavior definition method (Hornberger et al., 1980). This technique generates a set of parameter vectors on the basis of a pre-specified random distribution for each parameter. A simulation of the model is accomplished for each parameter vector. The model performance is then judged on the basis of knowledge of the system, or behavior definition (e.g., lower and upper bounds of state variables, distance of model trajectory from the observed values, etc.). This knowledge can be vague, allowing explicit incorporation of large uncertainties in the calibration procedure. With this technique is possible to account for uncertainties on model parameters and water quality measurements. Other inherent uncertainties (such as uncertainty in temperature, river streamflow etc.) could be taken into account in a sensitivity fashion (**see Chapter 3.8**).

Optimization unit

Usually, water quality management problems can be formulated as a search for suitable regional wastewater treatment policies. Many techniques have been applied to identify optimal water quality control strategies. The most commonly used are linear programming, dynamic programming, and simulation models. Non linear programming has been used more rarely due to its complexity and relevant computation requirements. DESERT uses dynamic programming (DP) technique (Bellman, 1957) for accomplishing the optimal allocation of waste water treatment plants. This technique decomposes a problem with a sequence of decisions into a cascade of subproblems, each having one or a reduced number of decisions. These subproblems are resolved recursively, by considering the sub-optimal solution(s) of one subproblem as input(s) to the subsequent subproblem.

The selection of optimal wastewater treatment alternatives in a river basin is a temporally and spatially sequential decision problem. Spatially, decisions are taken for a series of locations in a river basin. Due to the downstream propagation of pollutants of rivers, the water quality at a particular location along a river is entirely determined by the water quality at the immediate upstream discharge/control point (or at several discharge/control points in the case the location is below confluences). Similarly, when investigating investments over a temporal horizon, managers take decisions at points in time. Decisions taken at one time directly affect those to be taken at the next time step.

Clearly, the sequential attributes of DP make this approach very suitable for river water quality management.

Further, DP is generic - the solution algorithm does not depend on complexity of the model, linearity or non-linearity, number of state variables, etc.

One of the most usual formulation for allocating wastewater treatment plants is the following:

$$\min_{x_i \in \Omega_i} \sum_i C_i(x_i)$$

subject to: $WQ_{jk}(x_1, \dots, x_n) \leq WQS_{jk} \quad j, k$

where $C_i(x_i)$ is the cost function associated to the alternatives x_i

Ω_i is the set of possible alternatives for the object i

$Wq_{j,k}(x_1, \dots, x_n)$ is the value of the k th component of the water quality model produced by the set of alternatives x_1, \dots, x_n at the water quality constraint j

$WQS_{j,k}$ is the constrain on the component k of the water quality model to be satisfied at the water quality constraint j

Since the riverine system is treated as a binary tree, it is always possible to order the reaches as if they were located along a line starting from the most upstream point of the network and proceeding downstream. Therefore, it is possible to design a sequence of discharge/control points suitable for the DP approach. The water quality at the most upstream point is known (because it is a boundary condition), which implies that only one water quality state can be defined. A water quality state at the beginning of a reach is given by the value that components assume at that location. Of course, only the components that must satisfy some water quality constraint characterize states. The water quality resulting at the beginning of the downstream reach is estimated by using model predictions. The same procedure is then applied to the downstream reach, and so on proceeding downstream. When this procedure arrives to a Treatment plant object, the number of possible water quality states increases. The number of water quality states downstream of a treatment plant is given by the product of the water quality states

immediately upstream of the treatment plant times the number of alternatives characterizing the treatment plant itself. At a Water quality constraint point, the number of water quality states can be reduced by discarding the ones that do not meet the water quality requirements.

A screening operation at the beginning of each reach limits the growth of possible water quality states. States similar to others, but produced by a set of wastewater treatment alternatives having a higher value of the objective function, are discarded, and only the cheapest configurations are preserved for continuing the computation. Two states are considered similar when the distance (i.e., the difference in the value of components) between them is less than a specified quantity.

When the procedure reaches the root of the system (i.e., the End object), the set of feasible solutions (i.e., the solution satisfying the water quality requirements) is completed. Among these solutions, the one produced by the set of wastewater treatment alternatives with the smallest value of the objective function is the optimal solution. It is also possible to print all the feasible solutions in order to check the characteristics of the sub-optimal solutions.

This procedure is mathematically described by the following recursive form:
for the most upstream node

$$WQ_1 = M_1(x_1, WQ_0)$$

for the nth structural object

$$f_n(WQ_n) = \min_{\substack{x_n \in W_n \\ WQ_n \geq WQS_n}} (C_n(x_n) + f_{n-1}(WQ_{n-1}))$$

$$WQ_n = M_n(x_n, WQ_{n-1})$$

where $f_n(\dots)$ is the objective function at the nth structural object

$f_{n-1}(\dots)$ is the objective function at the n-1th structural object

$C_n(x_n)$ is the cost function associated to the management alternatives for the nth structural object (0 if no alternative is available)

W_n is the set of the possible management alternatives for nth structural object

WQ_n is the water quality at the downstream end of the nth reach

WQ_0 is the water quality at the most upstream node

WQS_n is the minimum water quality to be satisfied at the downstream end of the nth reach (0 if there is no requirement)

$M_n(x_n, WQ_{n-1})$ is the model prediction for the downstream end of the nth reach based on the water quality (WQ_{n-1}) and on the management alternative (x_n) at the upstream end of the reach.

The main drawback of DP technique is the rapid increase in memory requirements as the number of state variables increases (curse of dimensionality). Virtual memory in Windows 3.1 is a partial solution to this problem, but at the expense of computational speed

DESERT Window

When DESERT is invoked by double clicking on its icon with the mouse or by any other way allowed by Windows, the application window and the **TERMINAL** window will appear. The application window is a classical Windows style window, with its own title bar, menu bar, tool bar, bottom bar, and scroll bars. In the title bar there is the title, **Desert 1.0**, the minimizing and maximizing buttons, for resizing the window and reducing it to an icon, and the Control Menu button, for closing, restoring, minimizing, maximizing, sizing, and moving the window, and for switching to other tasks.

The menu bar contains a series of pop down menus

File

Edit

View

Window

Help

The tool bar is located immediately below the menu bar.

Tools

When a menu, a command, or a tool item are clicked with the pointer, a short description will appear on the bottom bar.

The menus can be accessed by using the mouse or by contemporarily typing the **Alt** digit and the digit corresponding to the letter underlined in the menu name.

File menu

The **F**ile menu contains all the commands necessary to create, open, save, import, and close files, and the command to exit from DESERT.

The menu can be displayed also by typing **Alt+F**.

The commands available in the **F**ile menu are:

New

Open

Inport

Save

Save As

Close

Exit

New

The **New** command causes the creation of a new Rivernet file. The created Rivernet file is initially empty. Data regarding a river network can be pasted in this file with the Paste command.

It can also be invoked by digit **Alt+F N**.

No more than one Rivernet file can be open at the same time. If a Rivernet file is open and the user tries to create a new Rivernet file, DESERT will signal an error.

Open

This command allows to open files **.RVR** and **.UNV** through a selection box. The Open command is not influenced by the window from which it is invoked. It always proposes the opening of a Universe file (.UNV).

However, by pressing the arrow button on the List file of Type box it is possible to switch from the default setting to the opening of a Rivernet file.

The result of the opening a file is the creation of a window and the graphical display of its content.

It can also be invoked by digit **Alt+F O**.

No more than one Rivernet file can be open att the same time. If a Rivernet file is open and the user tries to open anothe Rivernet file, DESERT will signal an error. On the contrary it is possible to open more than one Universe file at the same time.

Inport

This command allows to import an already defined Model file (.MOD). It is possible to invoke this command only from the Terminal window. When a Rivernet file window or a Universe file window is active, the command is not available and its instance in the File menu is grey. The result of invoking this command is the appearance of a dialogue box similar to the Open file dialogue box, but permitting the selection of MOD files only. It can also be invoked by digit **Alt+F I**.

Save

This command has a double effect according to which window it is invoked from.

If it is invoked from the Rivernet file window, it saves the content of this into a **.RVR** file with the same name of the window.

If it is invoked from the Terminal window, it saves the content of this in the text file Terminal.txt.

This command is not available when the Universe file window is active.

It can also be invoked by digit **Alt+F S**.

Save As

This command has characteristics similar to the Save command. However it allows to chose the name and the path of the .RVR file where to save the content of the Rivernet file window or the name and the path of the Text file where to save the content of the Terminal window.

It can also be invoked by digit **Alt+F A**.

See Also

[Open](#)

Close

This command closes the currently active window (files **.RVR** and **.UNV**). When the Terminal window is active, this command is not available. Consequently, it appears in gray color on the **File** menu.

It can also be invoked by digit **Alt+F C** or **Ctrl-F4**.

Exit

This command terminates the session of DESERT and closes all the currently open files. It can also be invoked by digit **Alt+F X** or **Alt+F4**.

Edit menu

The **Edit** menu contains all the commands used for editing (i.e., selecting, copying, and pasting) data files. Thus mainly **.UNV** and **.RVR** files are affected by its commands. However, the **Clear** command operates also on the Terminal window.

The menu can be displayed also by typing **Alt+E**.

The commands available in the Edit menu are:

Copy

Paste

Clear

Select Rivernet

Copy

This command is used to copy onto the clipboard a selection from the Universe file. The selection is obtained by using the **Select Rivernet** command.

It is available only when the Universe file window is active.

It can also be invoked by digit **Alt+E O** or **Ctrl+Ins**.

See Also

[Select Rivernet](#)

Paste

The selection from the Universe file copied with the **Copy** command can be pasted into a Rivernet file. Any new pasting will substitute the information contained in the Rivernet file with the new one. This substitution will become permanent if the file is saved.

It is available only when the Rivernet file window is active.

It can also be invoked by digit **Alt+E P** or **Shift+Ins**.

See Also

Copy

Clear

This command erases the content of the window from which it is invoked. When the Terminal window or the Rivernet file is cleared, DESERT does not have any information on the system or on the simulation to carry out. Thus it is necessary to substitute the erased information with a new one.

This command is not available when the Universe file window is active.

It can also be invoked by digit **Alt+E C** or **Ctrl+Del**.

Select Rivernet

This command allows to select a set of reaches from the **.UNV** file by a graphically encircling the reaches of interest. When this command is active, the open scissors symbol will appear on the screen in the place of the traditional arrow-shaped pointer. On the left upper corner of the scissors, there is a cross indicating the pointer of the mouse.

When the user has chosen the point from which the line enclosing the set of reaches of interest must begin, he or she must click the mouse. Since that moment a straight solid line departing from the point of clicking and going to the position of the pointer will be traced.

When the user will click the mouse again the straight line connecting the first clicking point to the second will become permanent. Another straight line will then connect the second clicking point to the pointer, allowing to trace another segment of the polygonal line necessary to delimit the region of interest. To select the reaches of interest, the polygonal line does not need to be exactly closed, but it needs to well define the desired area.

The drawing process is terminated by double clicking with the mouse. This action will black the area delimited by the polygonal line for few seconds. In this way it is possible to check whether the selected area is exactly the wished one.

It is to remember that the selected reaches must form a binary subtree. The selection of a set of not contiguous reaches will cause an error.

View menu

The **View** menu contains all the commands that characterize the representation of the Universe file on the computer screen. These commands can be used to facilitate viewing and accessing areas of interest.

The menu can be displayed also by typing **Alt+V**.

Fit to Window

Zoom Rectangle

Zoom in

Zoom out

Visibility

Fit to Window

This command displays the geometry of the system defined in the Universe file (and in the MapTable file, if it is used) at a window dimension. The smallest between the ratios width of the Universe file window/width of drawing and height of the Universe file window/height of the drawing is set equal to 2. The drawing is placed in the middle of the dimension having the smallest of the previous ratios.

This command can also be invoked by typing **Alt+V W**.


Zoom Rectangle

This command allows to zoom the vision of the geometry of the system to a dimension determined by the user through a flexible box. When invoked a cross shaped pointer appears in the place of the traditional arrow-shaped pointers (**see figure 4.5.2**). By dragging it, it is possible to enclose the area of interest in a flexible box. When the mouse is released, the area enclosed in the flexible box is displayed at the Universe file window dimension. However it is possible to access other parts of the Universe file by the scroll bars.

This command can also be invoked by typing **Alt+V R**.


Zoom in

This command magnifies twofold a portion of the screen.

When invoked, the  symbol appears in the place of the traditional arrow-shaped pointer. It serves as pointer for the center of the two fold decreasing of the scale of the vision.

Zoom out

This command zooms out by a factor of two a portion of the screen.

When invoked, the  symbol appears in the place of the traditional arrow-shaped pointer. It serves as pointer for the center of the two fold increasing of the scale of the vision.

Visibility

This command allows to hide the presence of some of the objects in the Universe file representation. This feature is very convenient, since in real life cases, hundreds of cross-sectional profiles can be necessary to represent a riverine system, while the number of other objects can easily arrive to the order of teens. This fact would make working on the Universe file representation very difficult. Representing only certain classes of objects makes the drawing less crowded of pictures and, consequently, handyer.

A dilaogue box allows the user to choose the classes of object that will be shown. The default configuration represents all the possible classes.

This commands can be invoked also by typing **Alt+V V**

Window menu

This menu contains the commands concerning the disposition of the windows used by DESERT (i.e., Terminal window, Universe file windows, and Rivernet file window) inside the DESERT application window.

The menu can be displayed also by typing **Alt+W**.

The commands available in the Menu commands are the following:

Cascade

Tile Horizontally

Tile Vertically

Arrange Icons

<#> <window name>

Cascade

This command arranges all the windows in a cascade style. It can be invoked also by typing **Alt+W C**.

Tile Horizontally

This command displays all the windows disposing them favoring width. That is the windows are disposed within the application window as horizontal bands. It can be invoked also by typing **Alt+W H**.

Tile Vertically

This command displays all the windows disposing them favoring height. That is the windows are disposed within the application window as vertical bands.

It can be invoked also by typing **Alt+W T**.

Arrange Icons

This command disposes document icons at the bottom of the screen.
It can be invoked also by typing **Alt+W I**.

<#> Window Name

This command fills the DESERT application window with the window characterized by the <Window name> name. It can be invoked also by typing **Alt+W <#>**, where <#> is the number associated to the window.

Help menu

The Help menu contains the commands for accessing to this Help file.
The menu can be displayed also by typing **Alt+H**.

About Desert

About Desert

This command displays information about this application.

Tools

The tool bar contains a series of buttons that allow to invoke the most commonly used commands in a quicker way.

The available tools are:



,



Working with DESERT 1.0

Working with DESERT requires following a series of steps from defining the input data sets to plotting the results.

- The first step involved is defining the input data files. Data files contain information such as the hydraulic description of the system, location and amount of emissions of waste load, points of measurements of water quality and quantity, etc.. Typically, such amount of heterogeneous information will be spread among several files and will have different formats. A particular text file, called the Universe file (extension **.UNV**), informs DESERT on how to retrieve the data from the appropriate files.

The input data files

- Once defined the characteristics of the system, you may desire to have a plot or to save the results of model simulations. In this case, you needs to establish a link between DESERT and a Spreadsheet.

Prepararing the link for exporting model results

- After these preliminary steps, DESERT can be invoked by double clicking the relative icon or by the other normal ways allowed by the Windows environment. It will appear a typical Windows application window. For the list of menus, command, and tools of the DESERT application

DESERT window

- Through the opening command, you will open the Universe file defined earlier. A representation of the systems as a binary tree will then appear. The tree will represent the reaches of the river system as well as the objects present in the system. Operating with graphical tools on the objects present on the tree, you will define the methods to be used to carry out the hydraulic and water quality simulation.

Working on the Universe file representation

- From the graphical representation of the Universe file, the reaches of interest will be moved to a new file, called the Rivernet file (extension **.RVR**). If a satisfactory file of this type was defined previously, then it may be open directly skipping the use of the Universe file. The Rivernet file supplies DESERT with the input data and the methods to be used to carry out both the hydraulic and the water quality simulation on the area of interest.

Definining a Rivernet file

Through the Terminal window, you must define the water quality model (i.e., variables, right hand sides of the transport equations, values of the constants, etc.) and give plotting and operational instructions. These operations are possible by writing the instructions directly on the Terminal window. A more practical alternative is writing first the instruction in a text file, called Model file (**.MOD**), through any text editor and then import this file to DESERT.

Defining the water quality models and running the simulations

- If the link with a spreadsheet has been established it is possible to export model results to it.

Exporting and plotting results

- Calibration and optimization of the river model will be accomplished through the

Terminal window also .

Calibration process

Optimization process

- An illustratory example of working with DESERT 1.0
Example

The input data files

When performing the analysis of the water quality of a river basin, data involved are very heterogeneous, ranging from the hydraulic properties of the system under study, to location and magnitude of waste emissions, to location and value of water quality measurements, etc.. Often data come from different sources and have different format.

Most of these data are vital for DESERT to carry out simulation, calibration, and optimization work. Others, although not directly necessary to DESERT, can be useful to you for a better understanding of the situation. Organizing the data in a format that could be used by the software and, at the same time, meet the variety of quantities and format of data that is typical of river basin management, is a challenging work. To achieve both these goals contemporarily, a powerful dBase style relational database engine has been introduced into DESERT.

You have only to hierarchically organize data files in two levels. The upper level consists of a text file informing DESERT on the organization of data files (**Universe** file).

The Universe file (.UNV)

The lower level groups the files actually containing information on the system:

- one .DBF file containing the list of the objects present in the system MasterTable file;
- one or more DBF files containing the bathymetry of the reaches and weir characteristics ProfileTable files;
- one .BNA file (optional) describing the geographical position of the structural objects MapTable file;
- one or more .DBF files containing lateral inflows, waste water discharges, water treatment alternatives, and the water quality measurements InfoTable files;
- one or more .DBF files (optional) containing the cost/benefits of the various alternatives for waste water treatment at the waste water treatment plants AlternativeTable files.

WARNING

Warning

When DESERT opens for the first time the .DBF files where information is stored, it creates a .NDX file for each .DBF file it opened. These index files are necessary to speed up the process of information retrieval by DESERT. Although they may occupy a not marginal spaces, they should not be canceled, otherwise a sensible slowing of the process will occur. However, if a change on the internal structure of any of the .DBF files has been made, the corresponding .NDX file must be canceled. If this precaution is not taken, a crash of the program may occur.

The Universe file (.UNV)

The universe file is a text file containing information on the organization of data regarding the system under study.

It is subdivided in sections, each of which groups the required information on a table file. Each of these sections is uniquely identified by a header indicating the name of the table file that is described. A further section, named General, describes some features that are common to several table files. The name of sections must be enclosed within brackets. Sections do not need to be written in a particular order, but, when there is the possibility to have several tables of the same type, their enumeration must be consequent. For example, if three InfoTables are used, their headers must be [InfoTable1], [InfoTable2], and [InfoTable3]. Using a not ordered enumeration like [InfoTable1], [InfoTable2], and [InfoTable4], will induce DESERT to skip the reading of the tables out of order. Also, sections need to satisfy the particular format that is given below.

In the following description, the word field relates to a database field of the file named at the beginning of the description of each section.

Description of the MasterTable file

[MasterTable]

Name=<path and name of the file containing the list of the objects defining the system>

RiverIndexField=<name of the field containing the River Index code>

CodeField=<name of the field containing the identification code of the objects>

OpCodeField=<name of the field containing the code indicating the type of object>

CISScale=<value with which multiplying the distances in the River Index to obtain the effective distances in meters>

CISLenght=<number of digits composing the CIS system (variable by nation)>

RoughnessField=<name of the field containing the Mannings roughness number of the river sections>

StyleField=<name of the field indicating an identification sub code of the type of object>

NameField=<name of the field containing the label to attribute to the objects>

General

[General]

Variant=<Default value for the management alternative. This name specifies the waste water treatment alternative that is used during normal simulation (i.e., when optimization is not performed)>

FlowField=<name of the field of the InfoTable and ProfileTable files that contains the measured discharge. This name field must be present for all the boundary condition and effluent specifications of all the InfoTables and for all the ProfileTables>

ElevationField=<name of the field of the InfoTable files that contains the water surface elevation of a water surface elevation-time relationship. The same name must be given to the field of the ProfileTable file that contains the

elevation part of the pairs bottom elevation-width describing the reach cross-sectional profile and of the pairs water elevation-discharge relationship >

Description of the Map Table file

[Map Table]

Name=<path and name of the file containing the Map information)

NumObjects=<the lateral size of the of the symbols in the system representation is set equal to the length of the shortest reach of the system divided by Numobjects. The default value is 3>

Description of the ProfileTable files

[ProfileTable #] where # is the number of the Profile Table (e.g., 1, 2, etc.)

Name=<path and name of the file containing the bathymetry of the river sections>

CodeField=<name of the field containing the identification code of the sections>

WidthField=<name of the field containing the width data>

Description of the InfoTable files

[InfoTable #] where # is the number of the InfoTable (e.g., 1, 2, etc.)

Name=<path and name of the file containing the InfoTable>

CodeField=<name of the field containing the identification code of objects>

DateField=<name of the field containing the date of measurements>

TimeField=<name of the field containing the time of measurements>

RadiusField=<name of the field containing the frequency of measurements>

VariantField=<name of the field containing the variant label>

Description of the AlternativeTable files (optional)

[AlternativeTable#] where # is the number of the AlternativeTable (e.g., 1, 2, etc.)

Name=<path and name of the file containing the AlternativeTable>

CodeField=<name of the field containing the identification code of the objects>

VariantField=<name of the field containing the variant label>

See also

[Example: example.UNV](#)

The MasterTable file

The Master Table File is a bBase III file listing all the elements present in the system and some of their characteristics. It may have as many fields as needed. However, DESERT utilizes only the following items:

<CodeField>

This is a code that uniquely identifies the objects. The code does not need to follow any standard, but it must be unique for any object present in the network. This field is composed by a not restricted number of chars.

<NameField>

This name will appear on the geographical representation of the data carried out by the display unit. This field is composed by a not restricted number of chars.

<RiverIndexField>

This is the location of the object given in the River Index Code.

<OpCodeField>

These codes inform DESERT of the type of objects it is dealing with. The available object codes for the version 1.0 of DESERT are the following:

Structural objects (See also Structural Objects)

'R' indicates the end of a reach and the beginning of another reach. Reaches are segments of the river uniform from some viewpoint (**see Chapter 1.1**).

'E' indicates the root of the basin system (End).

'C' indicates the confluence of two reaches. In the synthetic representation of a river system, the secondary reach will intersect the main branch on the right.

'CL' indicates the confluence of two reaches. In the synthetic representation of a river system, the secondary reach will intersect the main branch on the left.

'H' indicates the upstream extremities of the river system (Headwater).

River objects (See also River Objects)

- 'T' indicates a treatment plant. Water quality and discharge alternatives characterizing treatment plants are given in an InfoTable, while the alternatives costs are given in an AlternativeTable.
- 'I' indicates that along the reach where this object is located there is lateral inflow. Water quality and discharge characterizing the lateral inflow are given in an InfoTable.
- 'S' indicates a waste load point source. Water quality and discharge characterizing the source are given in an InfoTable.
- 'M' indicates a measurement point. Water quality and discharge characterizing the measurement point are given in an InfoTable.
- 'A' indicates an intake of water from a reach. Water quality and withdrawn discharge characterizing the intake are given in an InfoTable. However, DESERT will take into consideration only the discharge.
- 'Q' indicates a constrain in the water quality at any point. It is used to indicate that the water quality at a certain point needs to satisfy certain requirements. Water quality and discharge characterizing the water quality constrain are given in an InfoTable.
- 'F' indicates a cross section profile. Bathymetry regarding the cross section is given in a ProfileTable
- 'W' indicates the presence of a weir. A series of water elevation-flow pairs must be given in the ProfileTable, or a time-water elevation relationship must be described through an InfoTable.

WARNING

<StyleField>

The general characteristic of objects are indicated by their operational code. However, additional information may be sometimes necessary. The StyleField was introduced just to permit the definition of subsets of a main class of objects.

In the present version of DESERT, the StyleField is used in connection with River cross section objects and Weir. The River Cross sections are defined by a series of pairs width-elevation of the river bottom above the mean sea level. DESERT uses this information not only to calculate the cross-sectional area corresponding to a given water height, but also it uses the elevation of the bottom in the calculation of the slope of the reach. It may happen that, while the shape of the cross-section is reliable, its elevation above the mean sea level is not. In this case it may be useful to retain the shape of the profile, but the bottom elevation should not be used for the slope calculation. A 1 in the style field informs DESERT just of this situation.

The Weir objects can belong to two subclasses: the water stage elevation-discharge weirs (where a $Q(z)$ relationship is established) and the time-water stage weirs (where a $z(t)$ relationship is formulated). The first ones are individuated by QZ present on the style field. The latter ones by ZT in the Style field. In base of the value in the style field, DESERT will look either in the ProfileTable or in the InfoTable

for the data describing weirs behavior. This field is composed by a not restricted number of chars.

<RoughnessField>

This field contains the Mannings roughness number characterizing the river sections of the system. DESERT does not calibrate the hydraulic model of the system, but it only performs hydraulic simulation assuming as external data the value of cross sections, elevations, and roughness determined preliminarily using hydraulic models or table values. This field is a dBase float composed of n digits of which m<n are decimal. The size of m and n is chosen by you, taking into consideration the limits of the dBase language.

See also

[Example](#)

The River Index Code

The River Index Code is a way to describe a river system used in many countries. This method represents a river system as a binary tree. Therefore, each branch of the river system has a corresponding branch in the binary tree.

An order is associated to any branch. At every confluence of reaches in the binary tree it is possible to distinguish a main reach, that keeps the order of the downstream reach, and a secondary reach, that increases its order of 1. Thus, a position on the river system can be identified as a succession of distances composed by the number of digits indicated in the **CISLenght** specification of the Universe file.

If the object is situated on the main branch, then the first distance indicates the distance between the root of the tree and the object along this branch (of order 1) and the process terminates. If the object is not on the main branch, then the first distance is that between the root of the tree and the bifurcation where the path linking the object to the root leaves the main branch for a secondary branch (of order 2).

This process is recursively applied to the sub-tree constituted by the secondary branch and all its sub-branches. Consequently, the nth distance represents the distance between the root of the order nth and the object. If the object is located in a branch of order greater than n, then the nth is the distance between the root of the branch of order n and the bifurcation from which the branch of order n+1, supporting the sub-tree where the object is located departs. The process is then reapplied for the n+1th distance. The final result is a series of distances that uniquely identify the logical position of the object. You must remember that, on the contrary of what could be expected, the position of a confluence is always at the root of the secondary branch. This means that the River Index Code of a confluence terminates always with a value 0 expressed using as many digits as it is specified by the CISLenght specifier of the Universal file. The RiverIndexField is composed by a not restricted number of chars and the blocks of digits indicating the distances must not be separated from each other.

For instance, in the example system the objects EFFL0001, EFFL0002, HEAD0004, and CONF0002 have the following River Index codification:

```
EFFL0001: 003500
EFFL0002: 005000002500
HEAD0004: 00500000500005000
CONF0002: 010000000000
```

The MapTable file

The MapTable file contains the geographical coordinates of all structural objects. These data help DESERT in laying down a scheme closer to the real system.

The MapTable file is in the format of commercial Map Viewer (TM) mapping software from Golden Software Inc., although it can be easily produced manually from any text editor.

Each object needs two lines of description:

In the upper row, the first field must contain the identification code of the object between double quotation marks, the second field can contain its name, again between double quotation marks, eventually, the third field is a 1. The three fields are separated from each other by commas.

In the second row the two remaining fields, separated by a comma, will contain the geographical coordinates.

In the case the MapTable file is not specified, DESERT will proceed to draw a synthetic scheme of the river system, according to the logical position of objects given in the MasterTable.

In both cases, DESERT determines the ratio between height and width of the representation of the riverine system. DESERT then imposes that the smallest between the ratios width of the drawing/width of the Universe file window and height of the drawing/height of the Universe file window is equal to two (2), but at the same time preserving the ratio between the height and the width of the representation.

See also

[Example](#)

The ProfileTable files

A ProfileTable file is a .DBF file containing the bathymetry of the cross section profiles. Further, since weirs can be defined either in terms of a water stage-discharge relationship or in terms of a time-water stage relationship, the first type of relationship must be described in the ProfileTable, while the second should be described in a InfoTable. Profiles are defined by a series of pairs width (or distance from the longitudinal axis of the river)-elevation of the river bottom above mean sea level. Weirs characteristics are here defined as a series of pairs elevation-discharge. Thus each record of this file can contain either information on a cross-sectional profile or on a weir. Anyway, this file must be composed by at least the following fields:

<CodeField>

This field is composed by a not restricted number of chars. However, the size must equal that declared for the same field of the MasterTable file.

<WidthField

This is the width part of the width-elevation pair. In the case the record is describing a weirs water stage-discharge relationship, this field will not be considered. This field is a dBase float composed of n digits of which $m < n$ are decimal. The size of m and n is chosen by you, taking into consideration the limits of the dBase language.

<ElevationField>

This field can be either the elevation part of the width -elevation pair or the water stage part of a water stage elevation-discharge pair. This field is a dBase float composed of n digits of which $m < n$ are decimal. The size of m and n is chosen by you, taking into consideration the limits of the dBase language.

<FlowField>

This is the discharge part of a water stage-discharge pair. If the record keeps information on a cross-section profile, then this field will not be considered. This field is a dBase float composed of n digits of which $m < n$ are decimal. The size of m and n is chosen by you, taking into consideration the limits of the dBase language.

To completely define the profile of one river cross section, you must submit several width-elevation records regarding one of the halves of the cross-section profile (i.e., the space from the longitudinal axis to one of the sides). DESERT will complete the section by symmetry and linear interpolation. For defining weirs, you need to supply a series of water stage-discharge pairs. During the Steady state hydraulic solution, to compute the water stage DESERT will use the upstream discharge. The corresponding stage will be determined by linear interpolation of the supplied water stage-discharge pairs.

See also

Example

The InfoTable files

InfoTables are dBase III files containing very heterogeneous information on the system. Typically, they are used to store data on the water quality measurements of the effluent discharges into the system. They can also contain measurement data or constrains on the water quality of some reach of the system. However, they can be used to store water discharges or water stage measurements only. These values can be used as boundary conditions for the hydraulic and water quality simulation, for the calibration, and for the optimization tasks.

Since the number of water quality parameters measured at any site will vary from country to country and from project to project, their number has not been fixed. Of course, all the records of the same InfoTable will have to report the same number and type of data, but the possibility to use several InfoTables, allow a great flexibility. However, some fields need always to be present, such as:

<CodeField>

This field contains the identification code of the object. As specified for the other Table files, it can contain a not restricted number of chars, but it has to be declared of the same size of the analogous field in the MasterTable file.

<DateField>

This field contains the date of the measure. The date and time of measure are used when dynamic approximation of hydraulic or water quality equation is performed, in order to change boundary conditions or calibration matching with time. This field is of date type.

<TimeField>

This field contains the time of measure. It is composed by 5 chars. It plays the same role of the DateField.

<RadiusField>

This field contains the sampling time of measures. DESERT controls that the initial time and final time of simulation fall in the period of validity of available data. The period of validity is taken as <date and time of measure Sampling time/2>. In the case some parts of the simulation period are not covered by measured data, linear interpolation of the nearest available data is performed. The RadiusField is composed by one char. Available RadiusField codes are the following:

- H for hourly sampling
- D for daily sampling
- W for weekly sampling

M for monthly sampling

Y for yearly sampling

<VariantField>

This field contains the label indicating to what version the data belongs. It is mainly used for characterizing effluents of different water treatment schemes. Through the value of this field, the water quality records and the cost records regarding the same management alternative of wastewater treatment can be related to each other. It can be composed by a not limited number of chars.

<FlowField>

This field contains the measured discharge in m^3 / s . This field is a dBase float composed of n digits of which $m < n$ are decimal. The size of m and n is chosen by you, taking into consideration the limits of the dBase language. This field is necessary only for characterizing the objects that acts as upstream boundaries of the system (i.e., Headwater, Lateral inflow, Point sources, Treatment plant, Abstraction, and water stage-time Weir objects). For the other objects (i.e., Measurement and Water quality constrains, this field is not necessary

WARNING

See also
[example](#)

WARNING

The fields containing the other information can be of any type available in dBase. However, DESERT will automatically discard all the non numeric fields. To indicate that a numerical datum of one record is not reliable or available, all the digits of the relative field must be set to 9. The format (i.e., the total number of digits and the number of decimal digits) of the field must be respected. In this case DESERT is not taking the datum into consideration for the simulation. Data can be conveniently organized in different InfoTable files. In the example, the data relative to the water quality resulting from the different waste water treatment alternatives have been separated from those regarding the boundary conditions and water quality measures and constraints. It is also to be noticed that the two InfoTables do not contain the same fields. Because no heat budget was simulated in the example, the effluent water temperature of the water treatment plant is not necessary.

The AlternativeTable files

In the AlternativeTable file the cost of the waste water treatment alternatives needs to be specified. This file, which is also in a .DBF format, must contain also the following two fields:

<CodeField>

This field contains the identification code of the object. As specified for the other Table files, it can contain a not restricted number of chars, but it has to be declared of the same size of the analogous field in the MasterTable file.

<VariantField>

This field contains the label indicating to what version the data belongs. This field is mainly used for characterizing effluents of different water treatment schemes. It can be composed by a not limited number of chars.

The benefit/cost of the various alternatives must be specified in some kind of cost values that could be added. This is because at any stage of the dynamic programming method used to obtain the optimal allocation of resources, the cost of a feasible solution is obtained adding the cost of the feasible alternatives of the water treatment plant to the cost of the previous stage feasible solutions.

See also

[Optimization unit](#)

[Example](#)

Preparing the link for exporting model results

One of the most important steps in using simulation is evaluating model predictions. In doing this, scientists and managers need to study a large amount of data. Often this process is facilitated by the use of graphic representations. DESERT software allows to use directly the powerful means of spreadsheets to accomplish this task.

Although the OLE specifications used by DESERT are theoretically supported by many commercial spreadsheets, it turned out that, among the products tried, only Microsoft Excel for Windows version 4.0 and 5.0 have worked properly. Thus, it is strongly recommended the use of this popular software and all the present manual will make reference to it.

Some preliminary operations are necessary before running the system simulation if results have to be saved.

The first operation is to create an Excel worksheet large enough to contain the desired data (in the example this file is named **plotexe.xls**). This file must be filled with some sort of data. The kind of data is not important because DESERT will automatically erase them and substitute with the data that it produces. However, the number of data is important, since it fixes the size of the worksheet. Every simulation data that will not fit inside this worksheet will be discarded. This implicates that it is recommendable to create a very large file, composed of some hundreds of rows per some tens of columns, according to the possible amount of data to be saved.

See also

[Data transfer unit](#)

[Exporting and plotting results](#)

Working on the Universe file representation

After having defined the input data files and prepared the possible link to Microsoft Excel, DESERT can be invoked by double clicking its icon or by any other way guaranteed by Windows.

The DESERT window and the Terminal window will then appear. See [DESERT window](#)

When you want to define a new riverine system, change the methods of hydraulic and water quality simulation of a previously defined system, or when the input data file have been changed, you must open the Universe file describing the input data files. This operation can be done through the [Open](#) command of the [File](#) menu

When opening a .UNV file, DESERT organizes all the information sparse in the MasterTable, InfoTables, MapTables, ProfileTables, and AlternativeTables to eventually produce a map of the system and a symbolic representation of the objects.

All the operations DESERT performs to create this representation are echoed on the Terminal window. Possible errors in reading the input data file will be also indicated on this window.

Selecting an object through the mouse and double clicking on it introduces a dialogue box representing all the information available on that object. Through these dialogue boxes you might change the value of the data. However, changes will not affect the input data files, but only the internal representation of data created by DESERT. When the Universe file is closed, changes will vanish.

The Universe file representation has not only the passive role of showing the available information on the system, but also it is fundamental in selecting the parts of the system to be simulated and the methods by which performing the simulation. As introduced in [A general description of DESERT 1.0](#) one of the features that make DESERT an innovative software tool in environmental modeling is the possibility to use different solution methods for hydraulic and transport equations in different parts of the system. These methods are chosen by double clicking on any single reach of interest. A dialogue box will then appear for each reach that is double clicked.

Five methods for solving the hydraulic equations are available (for a theoretical explanation of the methods see [Hydraulic unit](#)):

[None](#)

[Water balance \(No mesh\)](#)

[Water balance \(Mesh\)](#)

[Steady state approximation](#)

[Diffusion wave approximation](#)

When any of the last three methods is chosen, a further dialog box will appear asking for the distance between consecutive mesh points. You will then specify the desired length. If the specified distance is not an exact divisor of the total length of the reach, DESERT will use as mesh size a value of its own choice. This value (which is possible to see by reopening the dialogue box) is determined according to the following scheme:

- the number of segments you desire is estimated by dividing the reaches length by the size you expressed;
- the mesh size used by DESERT is then set to the reach length divided by the number of

mesh segments determined at the first step. The final size will appear on the dialogue box whenever it is opened.

Although DESERT has not been designed for this purpose, it allows the simulation of hydraulics alone. In this case, there is no need to choose any solution method for the mass transport equations.

For modeling matter transport phenomena, six different approaches are available (for a theoretical background [Water quality simulation unit](#)).

None

Matter balance (No mesh)

Matter balance (Mesh)

Matter kinetics (No mesh)

Matter transport (Steady state approximation)

Matter transport (Dynamic approximation)

The choice of the method for performing hydraulic simulation influences also the choice of the method to solve the transport equations. If the None or the Water balance (with meshing or without meshing of the reach) methods are chosen, then, on the panel of water quality methods, only None and Matter balance (with meshing or without meshing of the reach) will appear.

On the other hand, by choosing the Steady state or the Diffusion wave approach for the hydraulics, the possibility to use Matter kinetic and Matter transport (either steady state or dynamic solution) will be also offered.

WARNING

The definition of the hydraulic and transport solution approaches is necessary only for the reaches that are of interest in the particular moment and not for all the system.

After this operation has been made, you must indicated to DESERT the part of the system that is the object of the simulation. This selection is carried out through a graphical method by using the [Select Rivernet](#) command. Selected reaches need to be connected and form a sub-tree of the system. They can not be isolated from each other, otherwise an error in the creation of the Rivernet file will be caused. In this case, a message will appear on the Terminal window advising that DESERT supports only one Rivernet.

See also

[Example](#)

Defining a Rivernet file

If you opened the Universe file and selected a river network as explained in the previous chapter, you must issue a New command. As a result of this action, an empty window named DESERT 1.0 - [Rivernet: river0001.RVR] will appear. You must now paste the previously selected network on this file by using the Paste command. The selected network will then appear on this window.

The Rivernet file defined in this way contains all the information that DESERT needs for carrying out the hydraulic and mass transport simulation. This file can be saved through the Save and Save As commands of the File menu.

When a Rivernet selection is saved as a .RVR file, the reaches and the parameters relative to the simulation procedures can be recalled directly without passing through the process of opening the Universe file, selecting the reaches, defining the simulation methods, and so on. The recall operation can be made simply by the Open command. Clearly, it is possible to open the Universe file, select a river network, open an existing Rivernet file, and paste the selection on this file. The previous content of the Rivernet file will be temporarily erased and substituted with the pasted data. This substitution will become permanent if the file is saved without changing its name.

When pasting a selection it may happen that the objects along a reach appear, but the link itself misses. This means that you forgot to define the hydraulic method for that reach. In this case, you must go back to the Universe file, define the hydraulic simulation for the missing reaches, and select, copy, and paste the area of interest again.

In the .RVR files, information stored in the objects are accessible and changeable as for the .UNV file. This is very useful in testing hypothesis regarding changes in the input data, without having the trouble to change the input data files. On the contrary of what happens with the Universe file, here changes data are saved when the Rivernet file is saved. However, changes can only affect input data. If the methods for performing hydraulic and water quality simulation need to be changed, then it is necessary to open the relative Universe file and repeat the operation described in the previous chapter. For this reason, double clicking a reach in the Rivernet file does not produce the same dialogue box that is produced in the Universe file. It will appear only a text box reporting the methods and the mesh size used in the reach.

By double clicking on any point of the white background of the .RVR file, a dialogue box will appear. It reports the time settings for the simulation of the hydraulic and water quality model. The Current date and the Current time refer, in this case, to the initial time and date of simulation. Finish date and time of simulation are also reported as well as the time step size in minutes. This data can be changed by you by writing on the relative windows.

See also

Example

Defining the water quality models and running the simulations

In order to run both the hydraulic model and the water quality model, you, after having obtained a .RVR file by creating it or by opening it, need to switch to the Terminal Window. Up to now, this window has played a passive role, being only the place where DESERT reported messages about errors or about the state of the process. At this point of the modeling procedure, this window assumes an active role, becoming the interface between DESERT and you for the definition of the water quality transport equations, the execution of the model, and the display of simulation results.

The language used to accomplish these tasks is called MODUS. It is conceptually similar to BASIC.

MODUS language explains in detail the glossary and syntax of this language.

DESERT interpreters and executes the commands written by you on the Terminal window. It also reports possible errors in the language syntax and in the execution of the simulation on the same window.

WARNING

Typing directly MODUS statements on the Terminal window can be convenient when you give execution commands (like **init**, **print**, **plot**, etc.) or define very simple water quality models. On the other hand, when models become complex, writing them every time is a really slow activity, where errors can be very frequent.

Also, the models written on the Terminal window can not be saved directly. It is only possible to save the content of the Terminal window as a text file (Save and Saves as commands). Working on this file it is possible to rescue the model, but it is not a very effective way of operating.

For solving this problem, DESERT allows the user to write the models in a text file through any text editor. The text file containing the model must be saved as a Model file (extension .MOD).

Then it can be imported to DESERT. Model files can be imported only when the Terminal window is active by the Import command. The model contained in the file is read, interpreted, and echoed on the Terminal Windows line by line. After that one line of the model has been read and echoed on the Terminal window, it is interpreted by DESERT. If this finds any syntax error, it will produce error messages on the lines below.

See also

Example

WARNING

DESERT is very case sensitive. If you have typed INIT, DESERT will signal an error of Invalid command. All the key words (**see Chapter 5**) of the MODUS language are in lower case letters. The case sensitiveness affects any letter written in the Terminal window. For example, the variables DO, dO, Do, and do are four distinct entities.

Exporting and plotting results

As already mentioned in [Preparing the link for exporting model results](#), to save results to a spreadsheet, a large Excel worksheet needs to be created.

This worksheet must be linked to DESERT for transferring results from DESERT to it. This operation is accomplished by pressing the [Link to Table](#) command from the Rivernet file window.

As a result of this operation, a typical openfile dialogue box will appear. The title of the box is Link To Table.

In the List Files of Type: box, the item Table (*.*) is present. This implicates that all the files of a directory are displayed in the box, independently from their file type

At this point you need to select the Excel file you created previously. Consequently, a dialogue box entitled Insert Object will appear. A choice of object types is here available. Anyway, you must choose the Microsoft Excel 5.0 Worksheet or the Microsoft Excel Worksheet option according to the version of Excel you are using.

After some seconds, a window entitled **Table001.TAB** (the number of the table depends on how many links you have established in the session) will appear. In the window the Excel file you defined earlier appears in a Clipboard style representation.

To establish the link between Excel and DESERT, you must double click inside this window. As a result of this, the Excel window of the Excel file you defined earlier will appear.

Every time the command **Plot** will be used while running the model, the values of the argument of this command will be automatically saved in this Excel file.

It is important you remember to save the data with a filename different from the original one (thus using the Excel command **Save as**). In this way, the format of the large file used for exchanging data will be saved. On the contrary, if data are saved with the original file name (using the Excel **Save** command), the format will be lost and a new large file will need to be constructed the next time saving data on a spreadsheet is required.

See also

[Example](#)

Calibration process

See also
[Example](#)

Optimization process

The ultimate objective of developing DESERT was to produce a software capable of optimally allocating waste water treatment plants. The method implemented in DESERT for achieving such allocation is the dynamic programming. Optimization unit illustrates the theoretical basis of the method and its implementation in DESERT. This chapter explains the procedure to work with the optimization procedure.

First of all, it must be clear that there is no theoretical limit on the number of treatment plants to allocate, the number of alternative solutions for each plant, and the number of constraints. However, if each of the treatment plants present on the system has one alternative only, than there is only one feasible solution. In this case, DESERT will determine only if it is compatible with the given constraints.

At the other extreme of the possible situations, there could be tens of treatment plants, each having several alternative implementations. In this case, if there are few constraints or if they are very loose, then the number of feasible alternatives grows dramatically. Although theoretically DESERT should be able to deal with the situation, the probable outcome would be a crash of the system due to the consumption of all the available memory. The boundary number of treatment plants, alternative implementations, and constraints depends on the hardware (i.e., processor type, RAM and hard disk space available, etc.) and software (i.e., number of tasks running concurrently, virtual memory size, etc.) configuration of the computing system.

Another factor to take into account in designing the optimization scheme is that the hydraulics in the riverine system is computed only once during optimization. The clear advantage of this choice is that the time for performing the optimization is strongly reduced. The drawback is that plants must have the same discharge in all the alternative schemes. This implicates that it is not possible to allocate directly position and size of effluent discharges, but only their quality. It is possible to overcome the problem by optimizing system representations where the treatment plants have different positions and then comparing the optimal solutions. However, this operation requires to set several configurations of the riverine system, optimize each of them, and compare the respective optimal solutions.

For performing the optimization, the user must specify the objective function and the possible constraints. Very likely, the objective function will have as arguments some kind of cost parameters associated to the possible alternative configurations of the treatment plants. For each water treatment plant several alternative configurations could be implemented. For each of them, the user must specify the cost parameters in an AlternativeTable file and the resulting quality of the effluent discharge in an InfoTable file. These two records are linked to each other by the values present in their CodeField and VariantField fields AlternativeTable files.

In the case more than one AlternativeTable file is used, the name of the fields containing the cost parameters must be the same.

In order to perform the optimization DESERT needs to know which fields of the AlternativeTable file are used as arguments of the objective function. This operation is accomplished by the statement:

property <identifier>

where **<identifier>** is the name of one field of the InfoTable containing one cost parameter. Such statement must be issued as many times as are the cost parameters. When this operation is done, the objective function must be specified by the statement:

objective <expression>

where **<expression>** is an expression having as possible arguments the cost parameters

specified by the **property** statements and the state variables. DESERT will allocate plants aiming to minimize the value of **<expression>**.

Water quality constraints must be expressed by statements of the type:

constraint <identifier> = <expression>

where **<identifier>** is the name given to the constraints and **<expression>** is an expression that will be evaluated as a C++ Boolean expression. This means that if **<expression>** has the value zero, then it is considered as false; it is considered as true otherwise.

See also

[Example](#)

Example

In order to facilitate the understanding of the use of DESERT, the construction of a model will be reproduced step by step. The model used as example is a trade off between simplicity and necessity of illustrating all the features of DESERT. For this reason, the numerical values used for the parameters are often not completely realistic.

The riverine system of the example (**figure 1**) is composed by a Main river, with two second order tributaries (SecondA and SecondB). SecondA has also a tributary of the third order (Tertiary). Almost every reach presents two Cross section objects, to allow the use of the Steady state and Diffusion wave approximation. However, the Tertiary river, does not show any Cross section object. On this reach, only the Water and Matter balance methods can be applied. This situation was designed just for demonstrating the possibility of using different hydraulic and water quality equations on different reaches. A Weir is also located on the upper branch of the SecondA river. The weir has the water stage elevation-discharge relationship given in **table 1**.

185.000	0.000
186.600	0.010
186.700	0.765
186.725	1.062
186.750	1.385
186.775	1.733
186.800	2.103
186.850	2.899
186.900	3.761

Table 1 Water stage elevation-discharge relationship for WEIR0001. The relationship has been derived using the formula for a broad-crested, well-ventilated weir, 13 m wide and 1.60 m high (White, 1986). However, to avoid stability problems during hydraulic dynamic simulation, a small discharge has been introduced for water stages lower than the crest of the weir.

The water quality simulation will regard a simple BOD-DO model. However, in order to give some background on the use of variables not subjected to the transport phenomena, an interaction with sediments has been introduced. Since the river system is artificial, the measurement data have been generated from the boundary conditions (**table 2**) using the model below:

$$\frac{\partial(A \times L)}{\partial t} + \frac{\partial(Q \times L)}{\partial x} = -K_d A \times L$$

$$\frac{\partial(A \times N)}{\partial t} + \frac{\partial(Q \times N)}{\partial x} = -K_d A \times N$$

$$\frac{\partial(A \times C)}{\partial t} + \frac{\partial(Q \times C)}{\partial x} = k_1 B \times (C_s - C) - K_d A \times L - K_d A \times N - B \times K_{sOD}$$

where: L - carbonaceous biological oxygen demand (CBOD) in mg / l

N - nitrogenous biological oxygen demand (NBOD) in mg / l

C - dissolved oxygen concentration in mg / l

x - coordinate along the river, m
t - travel time in days
Q - streamflow in m^3/s
A - cross-section area in m^2
B - stream width in m
 K_r - carbonaceous BOD removal rate in $1/\text{d}$
 k_a - oxygen exchange coefficient in $1/\text{d}$ (see OConnor and Dobbins, 1956)
 K_d - CBOD oxygenation rate in $1/\text{d}$,
 K_n - NBOD oxygenation rate in $1/\text{d}$,
 K_{SOD} - sediment oxygen demand in $\text{g}/\text{m}^2/\text{d}$
 C_s - saturation concentration of dissolved oxygen in mg/l

The parameter values used in producing the experimental data were:

$K_r = 0.8 \text{ 1/d}$
 $K_n = 0.8 \text{ 1/d}$
 $K_{\text{SOD}} = 1.5 \text{ g 2m / d}$
 $k_a = k_{a0} f(T) \sqrt{H}$

where: $ka_0 = 2.0 \text{ m}^{1/2} / \text{s}$
 U - flow velocity in m/ d
 H - aeration depth, defined as the ratio A/B

Figure 1 Description of the riverine system used as example.
The distances are intended from an object to another. However, cross-sectional profiles are not considered in determining distances (i.e., the upper part of the Main river has a total length of 7.5 km). Labels indicate the Code used for the objects.

Information on water discharges and quality (boundary conditions) are given in **table 2**.

HEAD0001	3.0	9.0	12.5	2.5	18
HEAD0002	1.0	10.0	10.0	2.0	15
HEAD0003	0.7	10.0	10.0	2.0	15
HEAD0004	0.3	10.83	5.0	1.5	12
LATE0001	0.01	7.0	15.0	20.0	15
	3m/s/km				
EFFL0001	0.150	1.0	250.0	48.0	22
EFFL0002	0.100	1.0	250.0	48.0	22

Table 2 Initial boundary conditions for the example (No treatment on EFFL0001 and EFFL0002).

The results of the model simulation, obtained by using steady state approximation for both hydraulic and water quality equations and the boundary conditions presented in **table 2**, are summarized in **table 3**.

CONF0001	16.74	3.44	4.81
CONF0002	8.86	2.04	6.18
CONF0003	8.01	1.74	9.53
END	12.08	2.48	2.80

Table 3 Measurements generated by the model used.

The system is characterized by the presence of two sources of untreated waste water: one in the middle reach of the Main river and the other in the first reach of the SecondA river. Two water quality constraints are present in the lower part of the first reach of SecondA and just upstream of the abstraction point. This situation is the minimum requirement for optimization of Water treatment plants' allocation. Water treatment alternatives are summarized in **table 4**.

No treatment (NT)	250	48	0
Primary (P)	175	40	0.064
Chemically Enhanced Primary Treatment (CEPT)	113	34	0.103

Primary Precipitation (PC)	75	34	0.122
Biological (B)	25	34	0.106
CEPT+B (BC1)	25	31	0.142
PC + B (BC2)	12	31	0.165
CEPT+B+partial Denitrification (BC1DN)	12	19	0.168
PC+B+partial Denitrification (BC2DN)	12	19	0.200
PC+B+Denitrification (BCDN)	7	7	0.210

Table 4 Waste water treatment alternatives for the pollution sources EFFL0001 and EFFL0002.

The files specifying the system used in the example are contained in the directory **desert\example**. They are the following:

example.unv

master.dbf

proftab.dbf

maptab.bna

infotab1.dbf

infotab2.dbf

altertab.dbf

plotfile.xls

bodnsod.mod

bnsodopt.mod

The example.unv file

The **example.UNV** is one Universe file describing the example system. This file asks DESERT to draw the system, according to the data contained in the file mapexe.bna. In order to have the synthetic generation of the system map, the MapTable must be skipped.

See also

[Display unit](#)

[MapTable file](#)

[MasterTable]

name=example\master.dbf

RiverIndexField=RIVERIND

CodeField=IDCODE

OpCodeField=OPCODE

CISScale=1

CISLength=6

SlopeField=SLOPE

RoughnessField=ROUGHM

StyleField=STYLE

NameField=NAME

[General]

Variant=1995

FlowField=Q

ElevationField=ELEV

[MapTable]

Name= example\maptab.bna

{This section must be skipped for
obtaining a synthetic generation of
the system representation}

[ProfileTable1]

Name= exampleproftab.dbf

CodeField=IDCODE

ElevationField=ELEV

WidthField=WIDTH

[InfoTable1]

Name=example\infotab1.dbf

CodeField=IDCODE

DateField=DATE

TimeField=TIME

RadiusField=PERIOD

VariantField=VARIANT

[InfoTable2]

Name=example\infotab2.dbf
CodeField=IDCODE
DateField=DATE
TimeField=TIME
RadiusField=PERIOD
VariantField=VARIANT

[AlternativeTable1]
Name=example/altertab.dbf
CodeField=IDCODE
VariantField=VARIANT

The master.dbf file

The organization of the MasterTable file used for the example (master.dbf).

MAIN0001	1st section on main	MAIN	000500	F	0.03000
MAIN0002	2nd section on main	MAIN	004500	F	0.03000
MAIN0003	3rd section on main	MAIN	005500	F	0.03000
MAIN0004	4th section on main	MAIN	009500	F	0.03000
MAIN0005	5th section on main	MAIN	010500	F	0.03000
MAIN0006	6th section on main	MAIN	017000	F	0.03000
SECA0001	1st section on SecondA	SECA	005000000500	F	0.03000
SECA0002	2nd section on SecondA	SECA	005000004500	F	0.03000
SECA0003	3rd section on SecondA	SECA	005000005500	F	0.03000
SECA0004	4th section on SecondA	SECA	005000009500	F	0.03000
SECB0001	1st section on SecondB	SECB	010000000500	F	0.03000
SECB0002	2nd section on SecondB	SECB	010000004500	F	0.03000
CONF0001	Confluence of SecondA on Main	MAIN	005000000000	MC	
CONF0002	Confluence of SecondB on Main	MAIN	010000000000	MC	
CONF0003	Confluence of Tertiary on SecondA	SECA	005000005000000000	MC	
HEAD0001	Headwater of Main	MAIN	017500	MH	
HEAD0002	Headwater of SecondA	SECA	005000010000	MH	
HEAD0003	Headwater of SecondB	SECB	010000005000	MH	

HEAD0004	Headwater of Tertiary	TERT	005000005000005000	MH	
END0001	Root of the system	MAIN	000000	ME	
EFFL0002	Effluent source on lower SecondA	SECA	005000002500	TR	
EFFL0001	Effluent source on intermediate Main	MAIN	007500	TR	
LATE0001	Lateral inflow on SecondB	SECB	010000002500	I	
ABST0001	Abstraction point on lower Main	MAIN	003000	A	
WQCO0001	Water quality constrain before the abstraction	MAIN	003500	QR	
WQCO0002	Water quality constrain on lower SecondA	SECA	005000001000	QR	
WEIR0001	Weir on the upper SecondA	MAIN	005000007500	WR	QZ

The proftab.dbf file

The ProfileTable used in the example (proftab.dbf).

MAIN0001	MAIN	000500	160.50000	0.00000
MAIN0001	MAIN	000500	163.50000	16.00000
MAIN0001	MAIN	000500	165.50000	22.00000
MAIN0002	MAIN	004500	164.50000	0.00000
MAIN0002	MAIN	004500	167.50000	15.00000
MAIN0002	MAIN	004500	169.50000	21.00000
MAIN0003	MAIN	005500	165.50000	0.00000
MAIN0003	MAIN	005500	168.50000	15.00000
MAIN0003	MAIN	005500	170.50000	21.00000
MAIN0004	MAIN	009500	169.50000	0.00000
MAIN0004	MAIN	009500	172.50000	14.00000
MAIN0004	MAIN	009500	174.50000	20.00000
MAIN0005	MAIN	010500	170.50000	0.00000
MAIN0005	MAIN	010500	173.50000	12.00000

MAIN0005	MAIN	010500	175.50000	18.00000
MAIN0006	MAIN	017000	177.00000	0.00000
MAIN0006	MAIN	017000	179.00000	7.00000
MAIN0006	MAIN	017000	182.00000	15.00000
SECA0001	SECA	000500	166.00000	0.00000
SECA0001	SECA	000500	168.00000	6.00000
SECA0001	SECA	000500	170.00000	12.00000
SECA0002	SECA	004500	174.00000	0.00000
SECA0002	SECA	004500	175.00000	6.00000
SECA0002	SECA	004500	178.00000	12.00000
SECA0003	SECA	005500	177.00000	0.00000
SECA0003	SECA	005500	178.00000	6.00000
SECA0003	SECA	005500	181.00000	11.00000
SECA0004	SECA	009500	193.00000	0.00000
SECA0004	SECA	009500	194.00000	5.00000
SECA0004	SECA	009500	197.00000	10.00000
SECB0001	SECB	000500	171.00000	0.00000
SECB0001	SECB	000500	172.00000	6.00000
SECB0001	SECB	000500	175.00000	11.00000
SECB0002	SECB	004500	179.00000	0.00000
SECB0002	SECB	004500	180.00000	7.00000
SECB0002	SECB	004500	183.00000	10.00000

The maptab.bna file

The maptab.bna file. This file must be used if a user supplied topography for the representation of the riverine is preferred to the synthetic representation.

"END0001","Downstream end of Main",1
16.00,13.00
"CONF0001","Confluence of SecondA on Main",1
16.00,15.00
"CONF0002","Confluence of SecondB on Main",1
16.00,17.00
"CONF0003","Confluence of Tertiary on SecondA",1
14.40,16.20
"HEAD0001","Headwater of Main",1
17.00,19.80
"HEAD0002","Headwater of SecondA",1
14.40,18.20
"HEAD0003","Headwater of SecondB",1
17.60,18.80
"HEAD0004","Headwater of Tertiary",1
12.40,16.20
"EFFL0001","Treatment plant on intermediate Main",1
16.00,16.00

"EFFL0002", "Treatment plant on lower SecondA", 1
15.10, 15.60

"WQCO0001", "Water quality before the abstraction", 1
16.00, 14.40

"WQCO0002", "Water quality constrain on lower SecondA", 1
15.68, 15.24

"WEIR0001", "Weir on the upper SecondA", 1
14.40, 17.20

The infotab1.dbf and infotab2.dbf files

Data can be conveniently organized in different InfoTable files. In the example, the data relative to the water quality resulting from the different waste water treatment alternatives have been separated from those regarding the boundary conditions and water quality measures and constraints. It is also to be noticed that the two InfoTables do not contain the same fields. Because no heat budget was simulated in the example, the effluent water temperature of the water treatment plant is not necessary.

HEAD0001	2/28/95	12:00	D	1995	3.000	9.0	12.5	2.5	18.0
HEAD0002	2/28/95	12:00	D	1995	1.000	10.0	10.0	2.0	15.0
HEAD0003	2/28/95	12:00	D	1995	0.700	10.0	10.0	2.0	15.0
HEAD0004	2/28/95	12:00	D	1995	0.400	10.8	5.0	1.5	12.0
LATE0001	2/28/95	12:00	D	1995	0.010	7.0	15.0	25.0	15.0
ABST0001	2/28/95	12:00	D	1995	0.100	0.0	0.0	0.0	15.0
CONF0001	2/28/95	12:00	D	1995	5.400	9999.9	9999.9	9999.9	17.7
CONF0002	2/28/95	12:00	D	1995	3.750	9999.9	9999.9	9999.9	17.4
CONF0003	2/28/95	12:00	D	1995	1.400	9999.9	9999.9	9999.9	14.0
WQCO0001	2/28/95	12:00	D	1995	5.400	5.5	11.3	2.3	18.0
WQCO0002	2/28/95	12:00	D	1995	1.500	5.5	11.3	2.3	15.0
END0001	2/28/95	12:00	D	1995	5.300	9999.9	9999.9	9999.9	18.0

Table 5 The InfoTable used in the example for boundary conditions and water quality constraints and measurements (Infoexe1.dbf).

EFFL0001	2/28/95	12:00	D	1995	0.450	1.0	250.0	48.0
EFFL0001	2/28/95	12:00	D	P	0.450	1.0	175.0	40.0
EFFL0001	2/28/95	12:00	D	CEPT	0.450	1.0	113.0	34.0
EFFL0001	2/28/95	12:00	D	PC	0.450	1.0	75.0	34.0
EFFL0001	2/28/95	12:00	D	B	0.450	1.0	25.0	34.0
EFFL0001	2/28/95	12:00	D	BC1	0.450	1.0	25.0	31.0
EFFL0001	2/28/95	12:00	D	BC2	0.450	1.0	12.0	31.0
EFFL0001	2/28/95	12:00	D	BC1DN	0.450	1.0	12.0	19.0
EFFL0001	2/28/95	12:00	D	BC2DN	0.450	1.0	12.0	19.0
EFFL0001	2/28/95	12:00	D	BCDN	0.450	1.0	7.0	7.0
EFFL0002	2/28/95	12:00	D	1995	0.300	1.0	250.0	48.0
EFFL0002	2/28/95	12:00	D	P	0.300	1.0	175.0	40.0
EFFL0002	2/28/95	12:00	D	CEPT	0.300	1.0	113.0	34.0
EFFL0002	2/28/95	12:00	D	PC	0.300	1.0	75.0	34.0
EFFL0002	2/28/95	12:00	D	B	0.300	1.0	25.0	34.0
EFFL0002	2/28/95	12:00	D	BC1	0.300	1.0	25.0	31.0
EFFL0002	2/28/95	12:00	D	BC2	0.300	1.0	12.0	31.0
EFFL0002	2/28/95	12:00	D	BC1DN	0.300	1.0	12.0	19.0
EFFL0002	2/28/95	12:00	D	BC2DN	0.300	1.0	12.0	19.0
EFFL0002	2/28/95	12:00	D	BCDN	0.300	1.0	7.0	7.0

Table 6 The InfoTable used in the example for the water quality of the different waste water treatment alternatives of the two water treatment plants (Infoexe2.dbf).

The altertab.dbf file

The benefit/cost of the various alternatives must be specified in some kind of cost values that could be added. This is because at any stage of the dynamic programming method used to obtain the optimal allocation of resources, the cost of a feasible solution is obtained adding the cost of the feasible alternatives of the water treatment plant to the cost of the previous stage feasible solutions.

EFFL0001	1995	0.000	0.0000
EFFL0001	P	0.064	0.0288
EFFL0001	CEPT	0.103	0.0464
EFFL0001	PC	0.122	0.0549
EFFL0001	B	0.106	0.0477
EFFL0001	BC1	0.142	0.0639
EFFL0001	BC2	0.165	0.0743
EFFL0001	BC1DN	0.168	0.0756
EFFL0001	BC2DN	0.200	0.0900
EFFL0001	BCDN	0.210	0.0945
EFFL0002	1995	0.000	0.0000
EFFL0002	P	0.064	0.0192
EFFL0002	CEPT	0.103	0.0309
EFFL0002	PC	0.122	0.0366
EFFL0002	B	0.106	0.0318
EFFL0002	BC1	0.142	0.0426
EFFL0002	BC2	0.165	0.0495
EFFL0002	BC1DN	0.168	0.0504
EFFL0002	BC2DN	0.200	0.0600
EFFL0002	BCDN	0.210	0.0630

Table 7 The AlternativeTable used in the Example (alterex1.dbf).

The measurement units for the cost have been added here for illustrative purposes and are not present in the actual file.

The plotfile.xls file

The plotfile.xls file is an Excel 5.0 workbook used for exporting results from DESERT. Its first worksheet is composed of 200 rows per 100 columns filled with 1. The main purpose of this file is to guarantee the existence of a Excel worksheet large enough to store exported data. In order to avoid the possibility to alter the file and save it changed (consequently loosing the format), plotfile.xls is a read-only file.

